

**PEMBANGUNAN SISTEM INFORMASI MANAJEMEN  
AKADEMIK SEKOLAH DASAR  
(STUDI PADA SDN WATES KABUPATEN KEDIRI)**

**SKRIPSI**

Untuk memenuhi sebagian persyaratan  
memperoleh gelar Sarjana Komputer

Disusun oleh:

Jawara Wahyu Al Faraday

NIM: 145150400111054



**PROGRAM STUDI SISTEM INFORMASI  
JURUSAN SISTEM INFORMASI  
FAKULTAS ILMU KOMPUTER  
UNIVERSITAS BRAWIJAYA  
MALANG  
2018**

## PENGESAHAN

PEMBANGUNAN SISTEM INFORMASI MANAJEMEN AKADEMIK SEKOLAH DASAR  
(STUDI PADA SDN WATES KABUPATEN KEDIRI)

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan  
memperoleh gelar Sarjana Komputer

Disusun Oleh :  
Jawara Wahyu Al Faraday  
NIM: 145150400111054

Skripsi ini telah diuji dan dinyatakan lulus pada  
17 Mei 2018

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II

Satrio Agung Wicaksono, S.Kom., M.Kom.  
NIP: 19860521 201212 1 001

M. Chandra Saputra, S.Kom., M.Eng.  
NIK: 201609 860106 1 001

Mengetahui  
Ketua Jurusan Sistem Informasi

Dr. Eng., Herman Tolle, S.T, M.T  
NIP: 19740823 200012 1 001

## PERNYATAAN ORISINALITAS

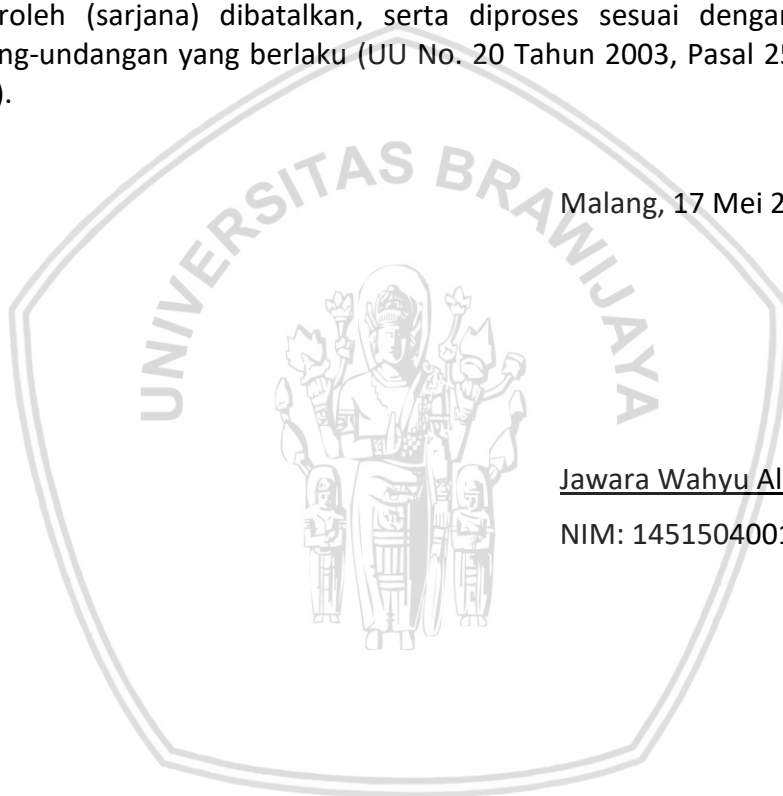
Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar pustaka.

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsur-unsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 17 Mei 2018

Jawara Wahyu Al Faraday

NIM: 145150400111054



## KATA PENGANTAR

Puji syukur penulis panjatkan kehadirat Allah SWT, karena berkat rahmat dan hidayah-Nya, penulis dapat menyelesaikan laporan skripsi yang berjudul “Pembangunan Sistem Informasi Manajemen Akademik Sekolah Dasar (Studi Pada SDN Wates Kabupaten Kediri)”.

Dalam penyusunan dan penulisan laporan ini penulis mendapatkan banyak bantuan dan dukungan dari berbagai pihak. Penulis ingin mengucapkan terimakasih yang sebesar - besarnya kepada :

1. Bapak Satrio Agung Wicaksono, S.Kom., M.Kom., selaku dosen Pembimbing 1 sekaligus dosen Penasehat Akademik yang senantiasa membimbing, memberikan motivasi dan saran kepada penulis selama penyusunan laporan skripsi.
2. Bapak Mochammad Chandra Saputra, S.Kom., M.Eng selaku dosen Pembimbing 2, yang telah membimbing, memberi saran serta motivasi kepada penulis selama penyusunan laporan skripsi.
3. Ibu Djumiasri, serta jajaran guru dan staf dari SDN Wates yang mengizinkan dan membantu penulis dalam menyelesaikan skripsi ini.
4. Bapak Wayan Firdaus Mahmudy, S.Si, M.T, Ph.D selaku Dekan Fakultas Ilmu Komputer Universitas Brawijaya Malang.
5. Bapak Herman Tolle, Dr. Eng., S.T, M.T selaku Ketua Jurusan Sistem Informasi Universitas Brawijaya Malang.
6. Bapak Suprpto, S.T, M.T selaku Ketua Program Studi Sistem Informasi Universitas Brawijaya Malang,
7. Bapak Aryo Pinandito, S.T, M.MT, Bapak Adam Hendra Brata, S.T, M.T, M.Sc, Bapak Ismiarta Aknuranda, S.T, M.Sc, Ph.D, Bapak Rahmat Trialih, S.Kom, M.Kom, Bapak Yusi Tyroni Mursityo S.Kom, M.AB, Ibu Retno Indah R. S.Pd, M.Pd, serta jajaran dosen yang telah memberikan saran dan motivasi kepada penulis.
8. Orang tua, kakak, serta saudara penulis atas segala dukungan yang telah diberikan kepada penulis.
9. Iqbal Taufiq, Mashuda Bahtiar, Danu Pramei, Januari Siregar, Dimas Primarizki, Aziza Zuhroh, Farhani, Firnanda, Cindy, Aya, Hana, Dika, Moi, Edwin, serta sahabat-sahabat penulis lainnya yang telah banyak memberikan motivasi dan semangat.
10. Sarah, Ain, Nathania, Ammar, Arda, serta adik-adik tingkat kece yang memberikan semangat kepada penulis.
11. Teman-teman Asisten Praktikum ADSI yang memberikan dukungan kepada penulis.

Penulis menyadari bahwa dalam penyusunan laporan skripsi ini masih banyak kekurangan baik format laporan maupun isinya. Untuk itu penulis sangat mengharapkan kritik dan saran yang membangun demi perbaikan selanjutnya. Semoga laporan skripsi ini dapat bermanfaat bagi pembaca dan penulis, amin.

Malang, 17 Mei 2018

Penulis

## ABSTRAK

Sekolah sebagai instansi pendidikan formal yang merupakan tempat untuk mencari ilmu, dalam kegiatan belajar mengajar. Sistem pendidikan nasional di Indonesia juga diatur dalam Undang-Undang Nomor 20, yang menyatakan bahwa pendidikan dasar merupakan prasyarat dasar untuk jenjang pendidikan yang lebih tinggi. SDN Wates merupakan sekolah dasar negeri yang berada di Kecamatan Wates Kabupaten Kediri. Namun terdapat beberapa masalah seperti penyebaran informasi yang dilakukan secara konvensional, proses pendaftaran dan seleksi murid yang masih dilakukan secara manual, proses evaluasi murid yang lama untuk melihat seluruh nilai mata pelajaran, serta pendataan alumni juga sulit dilakukan. Maka dari itu diperlukannya suatu sistem informasi yang dapat membantu menangani masalah tersebut. Proses pendaftaran dan seleksi murid, serta pengisian nilai dapat dipermudah dengan menggunakan Sistem Informasi Manajemen Akademik Sekolah Dasar Negeri Wates Kabupaten Kediri yang dibangun dengan metode *agile* dan diimplementasikan pada *website*, berdasarkan hasil pengujian *User Acceptance Testing* yang diberikan kepada pihak orang tua calon murid selaku pendaftar (nilai persentase UAT sebesar 85%), dan pihak guru selaku penyeleksi calon murid (nilai persentase UAT sebesar 80%), serta pihak guru selaku penilai murid (nilai persentase UAT sebesar 90%).

Kata kunci : Sistem Informasi, *Agile*, *User Acceptance Testing*, *Website*

## ABSTRACT

School as a formal educational institution which is a place to seek knowledge, in teaching and learning activities. The national education system in Indonesia is also regulated in Law No. 20, which states that basic education is a basic prerequisite for higher education. SDN Wates is a public elementary school located in Wates District Kediri District. However, there are several things such as the dissemination of manually-made information, the process of registration and refinement of data that is still done manually, the process for the old students to see all the value of the subjects, and also the data collection of alumni is also difficult. Therefore the need for information systems that can help deal with the problem. The process of enrollment and filling of students, as well as filling in the score can be simplified by using the Management Information System of the Primary School of Wates Kediri Regency built with agile method and implemented on the website, after the result of the User acceptance test given to the parents of the applicant students (the percentage of UAT by 85%), and the teacher student selection (UAT percentage score of 80%), as well as the teacher as a student appraiser (UAT percentage value of 90%).

Keywords : *Information System, Agile, User Acceptance Testing, Website*

## DAFTAR ISI

PENGESAHAN .....	ii
PERNYATAAN ORISINALITAS .....	iii
KATA PENGANTAR.....	iv
ABSTRAK.....	v
ABSTRACT .....	vi
DAFTAR ISI .....	vii
DAFTAR TABEL.....	xi
DAFTAR GAMBAR .....	xiii
DAFTAR LAMPIRAN .....	xv
BAB 1 PENDAHULUAN.....	1
1.1 Latar belakang.....	1
1.2 Rumusan masalah.....	2
1.3 Tujuan .....	3
1.4 Manfaat.....	3
1.5 Batasan masalah .....	3
1.6 Sistematika pembahasan.....	3
BAB 2 LANDASAN KEPUSTAKAAN .....	5
2.1 Kajian Pustaka .....	5
2.2 Sekolah Dasar .....	5
2.2.1 Sekolah Dasar Negeri Wates .....	6
2.3 Sistem Informasi .....	6
2.3.1 Sistem Informasi Manajemen .....	7
2.4 Rekayasa Perangkat Lunak ( <i>Software Engineering</i> ) .....	8
2.5 <i>Agile Software Development</i> .....	8
2.6 Proses Bisnis ( <i>Business Process</i> ) .....	10
2.6.1 <i>BPMN (Business Process Model and Notation)</i> .....	11
2.7 Analisis Kebutuhan Sistem.....	16
2.8 Perancangan Sistem.....	16
2.9 <i>UML (Unified Modelling Language)</i> .....	17
2.9.1 <i>Use Case Diagram</i> .....	18



2.9.2 Activity Diagram .....	19
2.9.3 Sequence Diagram .....	20
2.9.4 Class Diagram .....	21
2.9.5 Physical Data Model .....	22
2.10 Implementasi Sistem .....	23
2.10.1 Object Oriented Programming (OOP) .....	23
2.10.2 Model-View-Controller (MVC) Design Pattern .....	23
2.10.3 Website .....	24
2.11 Pengujian Sistem .....	24
2.11.1 Black-Box Testing .....	25
2.11.2 User Acceptance Testing (UAT) .....	25
2.12 Skala Likert .....	25
BAB 3 METODOLOGI PENELITIAN .....	27
3.1 Studi Literatur .....	27
3.2 Analisis Masalah & Pengumpulan Data .....	27
3.3 Mendefinisikan Kebutuhan .....	28
3.4 Analisis Inkremental .....	28
3.5 Pengembangan Pertama Sampai ke-N .....	28
3.6 Integrasi & Testing .....	28
3.7 Demo Rilis .....	29
3.8 Mengulas Umpan Balik .....	29
3.9 Formal Rilis .....	29
BAB 4 ANALISIS KEBUTUHAN .....	30
4.1 Aturan Penomoran .....	30
4.2 Pemodelan Proses Bisnis .....	31
4.2.1 Registrasi .....	31
4.2.2 Mengevaluasi Murid .....	32
4.3 Identifikasi Fitur .....	35
4.4 Identifikasi Aktor .....	36
4.5 Kebutuhan Fungsional .....	36
4.6 Kebutuhan Non-Fungsional .....	37
4.7 Use Case Diagram .....	37



4.8 Use Case Scenario .....	38
4.8.1 <i>Use Case Scenario</i> Mengisi Formulir Pendaftaran Murid (UC-SIAS-1) .....	38
4.8.2 <i>Use Case Scenario</i> Melakukan Verifikasi Pendaftaran (UC-SIAS-2) .....	39
4.8.3 <i>Use Case Scenario</i> Memasukkan Detail Nilai (UC-SIAS-5).....	40
4.8.4 <i>Use Case Scenario</i> Melihat Detail Nilai (UC-SIAS-6).....	40
4.8.5 <i>Use Case Scenario</i> Autentikasi (UC-SIAS-9).....	41
4.8.6 <i>Use Case Scenario</i> Keluar (UC-SIAS-10).....	42
BAB 5 PERANCANGAN DAN IMPLEMENTASI .....	43
5.1 Analisis Inkremental.....	43
5.2 Inkremen Pertama .....	43
5.2.1 <i>Activity Diagram</i> Mengisi Formulir Pendaftaran Murid (AD-SIAS-1) .....	44
5.2.2 <i>Sequence Diagram</i> Mengisi Formulir Pendaftaran Murid (SQ-SIAS-1) .....	45
5.2.3 Pemodelan <i>Class Diagram</i> .....	45
5.2.4 Pemodelan Data.....	46
5.2.5 Desain Algoritma Pendaftaran Murid .....	48
5.2.6 Desain Antarmuka Halaman Registrasi.....	50
5.2.7 Implementasi <i>Class Diagram</i> .....	51
5.2.8 Implementasi DDL ( <i>Data Definition Language</i> ) .....	51
5.2.9 Implementasi Fungsi Pendaftaran Murid .....	52
5.2.10 Implementasi Antarmuka Halaman Registrasi.....	54
5.3 Inkremen Kedua.....	55
5.3.1 <i>Activity Diagram</i> Melakukan Verifikasi Pendaftaran (AD-SIAS-2) .....	55
5.3.2 <i>Sequence Diagram</i> Melakukan Verifikasi Pendaftaran (SQ-SIAS-2) .....	56
5.3.3 Pemodelan <i>Class Diagram</i> .....	56
5.3.4 Pemodelan Data.....	57
5.3.5 Verifikasi Pendaftaran .....	61
5.3.6 Desain Antarmuka Halaman Verifikasi Pendaftaran.....	62

5.3.7 Implementasi <i>Class Diagram</i> .....	63
5.3.8 Implementasi DDL ( <i>Data Definition Language</i> ) .....	63
5.3.9 Implementasi Fungsi Verifikasi Pendaftaran .....	66
5.3.10 Implementasi Antarmuka Halaman Verifikasi Pendaftaran .....	67
5.4 Inkremen Ketiga .....	68
5.4.1 <i>Activity Diagram</i> Memasukkan Detail Nilai (AD-SIAS-5).....	68
5.4.2 <i>Sequence Diagram</i> Memasukkan Detail Nilai (SQ-SIAS-5).....	69
5.4.3 Pemodelan <i>Class Diagram</i> .....	69
5.4.4 Pemodelan Data .....	70
5.4.5 Desain Algoritma Memasukkan Detail Nilai.....	75
5.4.6 Desain Antarmuka Halaman Memasukkan Detail Nilai .....	77
5.4.7 Implementasi <i>Class Diagram</i> .....	78
5.4.8 Implementasi DDL ( <i>Data Definition Language</i> ) .....	78
5.4.9 Implementasi Fungsi Memasukkan Detail Nilai .....	82
5.4.10 Antarmuka Halaman Memasukkan Detail Nilai .....	86
BAB 6 PENGUJIAN .....	87
6.1 Pengujian <i>Black Box – Validation Testing</i> .....	87
6.1.1 Pengujian Fitur Mengisi Formulir Pendaftaran Murid .....	87
6.1.2 Pengujian Fitur Melakukan Verifikasi Pendaftaran .....	91
6.1.3 Pengujian Fitur Memasukkan Detail Nilai.....	93
6.1.4 Hasil Pengujian <i>Black Box – Validation Testing</i> .....	95
6.2 Pengujian <i>User Acceptance Testing</i> .....	95
6.2.1 Hasil Pengujian <i>User Acceptance Testing</i> .....	96
BAB 7 KESIMPULAN.....	98
7.1 Kesimpulan.....	98
7.2 Saran .....	98
DAFTAR PUSTAKA.....	99
LAMPIRAN A DOKUMENTASI WAWANCARA .....	100
LAMPIRAN B DOKUMENTASI UAT.....	104
LAMPIRAN C DOKUMENTASI VALIDASI BPMN .....	114

## DAFTAR TABEL

Tabel 2.1 Simbol-Simbol Dalam BPMN .....	11
Tabel 2.2 Notasi Dalam Diagram <i>Use Case</i> .....	18
Tabel 2.3 Notasi Dalam Diagram <i>Activity</i> .....	19
Tabel 2.4 Notasi Dalam Diagram <i>Sequence</i> .....	20
Tabel 2.5 Notasi Dalam Diagram <i>Class</i> .....	21
Tabel 2.6 Rincian Bobot Nilai .....	26
Tabel 2.7 Persentase Nilai .....	26
Tabel 4.1 Realisasi Aturan Penomoran .....	30
Tabel 4.2 Kodifikasi Aktivitas Proses Bisnis .....	35
Tabel 4.3 Identifikasi Fitur .....	35
Tabel 4.4 Identifikasi Aktor .....	36
Tabel 4.5 Kebutuhan Fungsional .....	36
Tabel 4.6 Kebutuhan Non-Fungsional .....	37
Tabel 4.7 Identifikasi <i>Use Case</i> .....	37
Tabel 4.8 <i>Use Case Scenario</i> Mengisi Formulir Pendaftaran Murid .....	38
Tabel 4.9 <i>Use Case Scenario</i> Melakukan Verifikasi Pendaftaran .....	39
Tabel 4.10 <i>Use Case Scenario</i> Memasukkan Detail Nilai .....	40
Tabel 4.11 <i>Use Case Scenario</i> Melihat Detail Nilai .....	40
Tabel 4.12 <i>Use Case Scenario</i> Autentikasi .....	41
Tabel 4.13 <i>Use Case Scenario</i> Keluar .....	42
Tabel 5.1 Tabel Analisis Inkremental .....	43
Tabel 5.2 Tabel Agama .....	47
Tabel 5.3 Tabel Pekerjaan .....	47
Tabel 5.4 Tabel Registrasi .....	47
Tabel 5.5 Algoritma Pendaftaran Murid .....	48
Tabel 5.6 Implementasi DDL .....	51
Tabel 5.7 Implementasi Fungsi Pendaftaran Murid .....	53
Tabel 5.8 Tabel Agama .....	58
Tabel 5.9 Tabel Pekerjaan .....	58
Tabel 5.10 Tabel Kelas .....	58

Tabel 5.11 Tabel Registrasi.....	59
Tabel 5.12 Tabel Orang Tua .....	59
Tabel 5.13 Tabel Murid .....	60
Tabel 5.14 Tabel Guru .....	60
Tabel 5.15 Algoritma Verifikasi Pendaftaran .....	61
Tabel 5.16 Implementasi DDL .....	63
Tabel 5.17 Implementasi Fungsi Verifikasi Pendaftaran.....	66
Tabel 5.18 Tabel Agama .....	71
Tabel 5.19 Tabel Pekerjaan .....	71
Tabel 5.20 Tabel Kelas.....	71
Tabel 5.21 Tabel Registrasi.....	72
Tabel 5.22 Tabel Orang Tua .....	72
Tabel 5.23 Tabel Murid .....	73
Tabel 5.24 Tabel Guru .....	73
Tabel 5.25 Tabel Tahun Ajaran .....	74
Tabel 5.26 Tabel Semester .....	74
Tabel 5.27 Tabel Jenis .....	74
Tabel 5.28 Tabel Mata Pelajaran .....	74
Tabel 5.29 Tabel Nilai .....	74
Tabel 5.30 Algoritma Memasukkan Detail Nilai.....	75
Tabel 5.31 Implementasi DDL .....	78
Tabel 5.32 Implementasi Fungsi Memasukkan Detail Nilai .....	82
Tabel 6.1 Kasus uji mengisi formulir pendaftaran murid: <i>basic flow</i> .....	87
Tabel 6.2 Kasus uji mengisi formulir pendaftaran murid: <i>alt1</i> .....	89
Tabel 6.3 Kasus uji melakukan verifikasi pendaftaran: <i>basic flow</i> .....	91
Tabel 6.4 Kasus uji memasukkan detail nilai: <i>basic flow</i> .....	93
Tabel 6.5 Pengujian UAT .....	95

## DAFTAR GAMBAR

Gambar 2.1 Sistem Informasi.....	7
Gambar 2.2 Klasifikasi Tipe Sistem Informasi .....	8
Gambar 2.3 Biaya Perubahan dalam Pengembangan .....	10
Gambar 2.4 Contoh penggambaran diagram proses bisnis.....	15
Gambar 2.5 Contoh penggambaran diagram <i>Use Case</i> .....	18
Gambar 2.6 Contoh penggambaran diagram <i>Activity</i> .....	20
Gambar 2.7 Contoh diagram <i>Sequence</i> .....	21
Gambar 2.8 Contoh diagram <i>Class</i> .....	22
Gambar 2.9 Contoh <i>physical data model</i> .....	23
Gambar 3.1 Alur Penelitian .....	27
Gambar 4.1 Proses Bisnis Registrasi Saat Ini.....	31
Gambar 4.2 Proses Bisnis Registrasi Usulan .....	32
Gambar 4.3 Proses Bisnis Mengevaluasi Murid Saat Ini.....	33
Gambar 4.4 Proses Bisnis Mengevaluasi Murid Usulan.....	34
Gambar 4.5 <i>Use Case Diagram</i> .....	38
Gambar 5.1 <i>Activity Diagram</i> Mengisi Formulir Pendaftaran Murid .....	44
Gambar 5.2 <i>Sequence Diagram</i> Mengisi Formulir Pendaftaran Murid .....	45
Gambar 5.3 <i>Class Diagram</i> Home sebagai <i>Logical Class</i> .....	46
Gambar 5.4 <i>Class Diagram</i> Registrasi_model sebagai <i>Domain Model</i> .....	46
Gambar 5.5 <i>Physical Data Modeling</i> .....	47
Gambar 5.6 <i>Flowchart</i> Pendaftaran Murid.....	49
Gambar 5.7 Desain Antarmuka Halaman Registrasi .....	50
Gambar 5.8 Implementasi <i>Class Diagram</i> .....	51
Gambar 5.9 Antarmuka Halaman Registrasi.....	54
Gambar 5.10 <i>Activity Diagram</i> Melakukan Verifikasi Pendaftaran .....	55
Gambar 5.11 <i>Sequence Diagram</i> Melakukan Verifikasi Pendaftaran.....	56
Gambar 5.12 <i>Class Diagram</i> Guru sebagai <i>Logical Class</i> .....	57
Gambar 5.13 <i>Class Diagram</i> sebagai <i>Domain Model</i> .....	57
Gambar 5.14 <i>Physical Data Modeling</i> .....	58
Gambar 5.15 <i>Flowchart</i> Verifikasi Pendaftaran.....	61

Gambar 5.16 Desain Antarmuka Halaman Verifikasi Pendaftaran.....	62
Gambar 5.17 Implementasi <i>Class Diagram</i> .....	63
Gambar 5.18 Antarmuka Halaman Verifikasi Pendaftaran.....	67
Gambar 5.19 <i>Activity Diagram</i> Memasukkan Detail Nilai .....	68
Gambar 5.20 <i>Activity Diagram</i> Memasukkan Detail Nilai .....	69
Gambar 5.21 <i>Class Diagram</i> Guru sebagai <i>Logical Class</i> .....	70
Gambar 5.22 <i>Class Diagram</i> sebagai <i>Domain Model</i> .....	70
Gambar 5.23 <i>Physical Data Modeling</i> .....	71
Gambar 5.24 <i>Flowchart</i> Memasukkan Detail Nilai .....	76
Gambar 5.25 Desain Antarmuka Halaman Memasukkan Detail Nilai .....	77
Gambar 5.26 Implementasi <i>Class Diagram</i> .....	78
Gambar 5.27 Antarmuka Halaman Memasukkan Detail Nilai .....	86
Gambar 6.1 Kasus uji mengisi formulir pendaftaran murid: <i>basic flow</i> .....	88
Gambar 6.2 Kasus uji mengisi formulir pendaftaran murid: <i>basic flow</i> .....	88
Gambar 6.3 Kasus uji mengisi formulir pendaftaran murid: <i>basic flow</i> .....	89
Gambar 6.4 Kasus uji mengisi formulir pendaftaran murid: alt1 .....	90
Gambar 6.5 Kasus uji mengisi formulir pendaftaran murid: alt1 .....	90
Gambar 6.6 Kasus uji mengisi formulir pendaftaran murid: alt1 .....	90
Gambar 6.7 Kasus uji melakukan verifikasi pendaftaran: <i>basic flow</i> .....	91
Gambar 6.8 Kasus uji melakukan verifikasi pendaftaran: <i>basic flow</i> .....	92
Gambar 6.9 Kasus uji melakukan verifikasi pendaftaran: <i>basic flow</i> .....	92
Gambar 6.10 Kasus uji melakukan verifikasi pendaftaran: <i>basic flow</i> .....	93
Gambar 6.11 Kasus uji memasukkan detail nilai: <i>basic flow</i> .....	94
Gambar 6.12 Kasus uji memasukkan detail nilai: <i>basic flow</i> .....	94
Gambar 6.13 Kasus uji memasukkan detail nilai: <i>basic flow</i> .....	94
Gambar 6.14 Kasus uji memasukkan detail nilai: <i>basic flow</i> .....	95

## DAFTAR LAMPIRAN

LAMPIRAN A DOKUMENTASI WAWANCARA .....	100
A.1 Wawancara dengan Kepala SDN Wates .....	100
LAMPIRAN B DOKUMENTASI UAT.....	104
LAMPIRAN C DOKUMENTASI VALIDASI BPMN .....	114





## BAB 1 PENDAHULUAN

### 1.1 Latar belakang

Seiring berjalannya waktu, teknologi informasi dan komunikasi terus berkembang dan semakin pesat, sehingga menjadi suatu kebutuhan yang umum. Internet merupakan salah satu bentuk dari berkembangnya teknologi informasi dan komunikasi yang telah merambah ke segala aspek dan bidang dalam kehidupan pribadi maupun global. Demikian sama halnya dengan dunia pendidikan yang terus mengalami perubahan menuju kearah yang lebih baik lagi. Untuk tercapainya perubahan menuju kearah yang lebih baik lagi, tentunya tidak lepas dari pengaruh penggunaan teknologi informasi dan komunikasi.

Sekolah sebagai instansi pendidikan formal yang merupakan tempat untuk mencari ilmu, dalam kegiatan belajar mengajar (Suharjo, 2006). Sistem pendidikan nasional di Indonesia juga diatur dalam Undang-Undang Nomor 20, yang menyatakan bahwa pendidikan dasar merupakan prasyarat dasar untuk jenjang pendidikan yang lebih tinggi. Hingga saat ini sekolah terus berusaha menjadi instansi pendidikan yang mampu melayani masyarakat dengan baik, terutama dalam hal penyebarluasan informasi kepada masyarakat umum. Penulis memilih SDN Wates sebagai objek penelitian karena terdapat beberapa permasalahan diantaranya seperti yang diutarakan menurut Ibu Djumiasri selaku kepala SDN Wates yaitu: penyebarluasan tentang informasi sekolah beberapa masih dilakukan secara konvensional, yakni kebanyakan informasi terbaru ditempel di papan pengumuman, surat undangan yang dibagikan kepada siswa maupun orang tua siswa dan masih menggunakan media informasi seperti poster sebagai media promosi kepada masyarakat umum. Proses pendaftaran murid masih dilakukan secara manual, yaitu dengan mengambil formulir untuk diisi, kemudian diserahkan kembali pada pihak sekolah untuk diseleksi. Pada proses evaluasi, murid harus menunggu proses penerimaan raport terlebih dahulu untuk melihat semua nilai dari keseluruhan mata pelajaran. Selain itu, pendataan alumni juga sulit dilakukan. Maka dari itu diperlukannya suatu sistem informasi yang dapat membantu menangani masalah tersebut.

Pihak SDN Wates menginginkan pembangunan sistem informasi yang cepat, dan mengakomodir kebutuhan. Maka penulis memilih metode *agile* sebagai metode pembangunan sistem informasi.

Dalam pembangunan suatu sistem informasi, hal yang pertama dilakukan adalah pendefinisian masalah, berlanjut ke analisis kebutuhan. Mengacu pada standar IEEE nomor: *IEEE-STD-1220-1998*, kebutuhan adalah sebuah pernyataan yang mengidentifikasi produk atau proses, baik secara operasional maupun fungsional. Setelah memastikan kebutuhan telah dianalisis dengan benar, tahap selanjutnya yaitu melakukan perancangan sistem. Tahap perancangan merupakan tahap yang penting dalam pengembangan suatu sistem karena kualitas hasil perancangan dapat mempengaruhi kualitas sistem. Perancangan yang mudah dalam pemeliharaan adalah perancangan dengan pendekatan *Object Oriented*

(Sommerville, 2011). Setelah melalui tahap ini, hasil rancangan diwujudkan dalam bentuk sistem yang berfungsi secara utuh, menggunakan bahasa pemrograman. Bahasa pemrograman memiliki 2 orientasi yaitu terstruktur dan *Object Oriented*. Untuk sistem yang dirancang dengan pendekatan *Object Oriented* menggunakan *Object Oriented Programming*, dan diimplementasikan dalam sistem berbasis *website* untuk kemudahan aksesnya. Terakhir, pengujian digunakan untuk memastikan sistem berjalan sesuai rancangan dan memenuhi semua kebutuhan yang telah di definisikan. Beberapa metode pengujian yang dapat dilakukan yakni *Black Box – Validation Testing* untuk memastikan sistem berjalan sesuai dengan skenario (Homes, 2012), serta *User Acceptance Testing* untuk mengetahui penerimaan sistem terhadap pengguna.

Penelitian sebelumnya yang dilakukan oleh Sugiarti & Wardati pada tahun 2012, dengan beberapa masalah yang melatarbelakangi penelitian tersebut yaitu proses pendataan siswa, guru, mata pelajaran beserta nilainya yang masih secara konvensional sehingga mengakibatkan keterlambatan dalam penyajian laporan. Keterkaitan penelitian tersebut dengan penelitian yang dilakukan penulis ialah dalam bidang yang sama, yakni bidang akademik siswa sekolah dasar. Penelitian selanjutnya yang dilakukan oleh Shafirna pada tahun 2017 membahas mengenai analisis dan perancangan sistem informasi manajemen berbasis *website* untuk kelompok bermain dan taman kanak-kanak Permata Iman. Masalah yang melatarbelakangi penelitian tersebut yaitu proses penilaian evaluasi belajar siswa yang masih terbelang konvensional, guru yang menginginkan proses pembelajaran yang sejalan antara siswa belajar di sekolah dengan belajar di rumah. Keterkaitan penelitian tersebut ialah menggunakan orientasi yang sama dalam perancangan sistemnya, yaitu menggunakan *object oriented design*. Penelitian selanjutnya yang dilakukan oleh Ependi pada tahun 2012 membahas tentang pendataan alumni secara lokal (*stay alone*) yang menyebabkan sulitnya pendistribusian data alumni terhadap perusahaan rekanan, sehingga melatarbelakangi pengembangan sistem.

Berdasarkan uraian pernyataan tersebut, peneliti tertarik untuk melakukan penelitian tentang pembangunan sistem informasi manajemen akademik sekolah, yang nantinya akan diterapkan di SDN Wates Kabupaten Kediri. Judul penelitian yang diambil oleh peneliti yakni “Pembangunan Sistem Informasi Manajemen Akademik Sekolah Dasar (Studi Pada SDN Wates Kabupaten Kediri)”. Melalui penelitian ini, diharapkan dapat diimplementasikan guna mempermudah proses pengelolaan data, serta tercukupinya kebutuhan masyarakat umum akan layanan pendidikan yang aktual dan informatif.

## 1.2 Rumusan masalah

Berdasarkan latar belakang yang telah di uraikan, rumusan masalah pada penelitian ini sebagai berikut:

1. Bagaimana mempermudah proses pendaftaran dan verifikasi pendaftaran murid pada Sekolah Dasar Negeri Wates Kabupaten Kediri?

2. Bagaimana mempermudah proses penyampaian nilai (rapor) dengan menggunakan Sistem Informasi Manajemen Akademik Sekolah Dasar Negeri Wates Kabupaten Kediri?

### 1.3 Tujuan

Adapun tujuan yang ingin dicapai dalam penelitian ini, ialah:

1. Mempermudah proses pendaftaran dan verifikasi pendaftaran murid.
2. Mempermudah proses penyampaian nilai (rapor) dengan menggunakan Sistem Informasi Manajemen Akademik Sekolah Dasar Negeri Wates Kabupaten Kediri.

### 1.4 Manfaat

Manfaat yang diharapkan didapat dari penelitian ini ialah sebagai berikut:

1. Bagi Penulis
  - a. Menambah pengetahuan mengenai cara untuk mempermudah pendaftaran dan verifikasi pendaftaran murid.
  - b. Menambah pengetahuan mengenai mempermudah proses penyampaian nilai (rapor) dengan menggunakan Sistem Informasi Manajemen Akademik Sekolah Dasar Negeri Wates Kabupaten Kediri.
2. Bagi Pembaca
  - a. Memberikan wawasan mengenai pendaftaran dan verifikasi pendaftaran murid.
  - b. Memberikan wawasan mengenai mempermudah proses penyampaian nilai (rapor) dengan menggunakan Sistem Informasi Manajemen Akademik Sekolah Dasar Negeri Wates Kabupaten Kediri.

### 1.5 Batasan masalah

Penelitian ini memiliki batasan masalah sebagai berikut:

1. Data yang diperoleh merupakan hasil wawancara langsung dengan pihak Sekolah Dasar Negeri Wates Kabupaten Kediri.
2. Metode yang digunakan untuk membangun sistem informasi adalah metode *Agile*.
3. Desain sistem yang dilakukan adalah desain sistem berbasis *Object Oriented*, menggunakan bahasa *UML*.
4. Penelitian ini dilakukan mulai analisis kebutuhan, perancangan, implementasi, hingga pengujian sistem.

### 1.6 Sistematika pembahasan

Sistematika bahasan dalam penyusunan skripsi ini ialah sebagai berikut::

**BAB 1 PENDAHULUAN**

Bab ini berisi penjelasan tentang latar belakang masalah yang ada, rumusan masalah, tujuan dari penelitian, manfaat yang diharapkan didapat, batasan masalah serta sistematika penulisan yang di gunakan.

**BAB 2 LANDASAN KEPUSTAKAAN**

Bab ini menjelaskan tentang dasar teori yang digunakan dalam mendukung topik penelitian. Bab ini juga memuat kajian pustaka yangmana terdapat referensi skripsi maupun jurnal dengan penelitian yang relevan atau hampir sama pada bidangnya.

**BAB 3 METODOLOGI PENELITIAN**

Pada bab ini, menjelaskan tentang langkah – langkah yang di tempuh atau metode penelitian yang digunakan dalam melakukan penelitian ini. Metode yang digunakan memuat pengambilan data, pengembangan sistem, pengolahan data serta solusi yang akan diberikan dari hasil penelitian ini.

**BAB 4 ANALISIS KEBUTUHAN**

Pada bab ini menjelaskan pengidentifikasian kebutuhan apa saja baik kebutuhan fungsional maupun kebutuhan non-fungsional untuk sistem yang akan dibangun berdasarkan metodologi penelitian.

**BAB 5 PERANCANGAN DAN IMPLEMENTASI**

Bab ini menjelaskan bagaimana memodelkan kebutuhan kedalam rancangan (*blue print*) berupa diagram perancangan sistem, sesuai dengan metode yang digunakan. Perancangan ini berorientasi pada *object*. Serta realisasi dari rancangan sistem yang ditulis menggunakan bahasa pemrograman php. Metode yang diterapkan dalam penelitian ini yaitu menggunakan metode *agile (iteratif - incremental)*.

**BAB 6 PENGUJIAN**

Pada bab ini menjelaskan bagaimana sistem yang telah dibangun akan diuji berdasarkan metode pengujian *black box – validation testing* dan metode pengujian *user acceptance testing*.

**BAB 7 PENUTUP**

Bab ini menjelaskan simpulan berdasar keseluruhan penelitian beserta saran yang diberikan terhadap hasil penelitian yang telah dilakukan.

## BAB 2 LANDASAN KEPUSTAKAAN

Bab ini memuat dasar teori dan kajian pustaka yang akan digunakan peneliti untuk mendukung proses penelitian. Kajian pustaka berisi penelitian terkait yang sebelumnya sudah dilakukan, digunakan sebagai acuan pada penelitian ini, dan dasar teori akan dijadikan pedoman untuk melakukan penelitian.

### 2.1 Kajian Pustaka

Isi dari kajian pustaka berupa beberapa penelitian terkait yang telah dilakukan sebelumnya. Penelitian pertama yang dilakukan oleh Sugiarti & Wardati pada tahun 2012, terdapat beberapa masalah yang melatarbelakangi penelitian tersebut ialah proses pendataan siswa, guru, mata pelajaran beserta nilainya yang masih secara konvensional sehingga mengakibatkan keterlambatan dalam penyajian laporan. Dalam penelitian tersebut, sistem informasi dibangun dengan memodelkan proses menggunakan *DFD (Data Flow Diagram)*, memodelkan data dengan *ERD (Entity Relationship Diagram)*, menggunakan *MySQL DBMS* untuk pengelolaan data, dan menggunakan *PHP* sebagai bahasa pemrograman web. Keterkaitan penelitian tersebut dengan penelitian yang dilakukan penulis ialah dalam bidang yang sama, yakni bidang akademik siswa sekolah dasar.

Penelitian kedua yang dilakukan oleh Shafirna pada tahun 2017 membahas mengenai analisis dan perancangan sistem informasi manajemen berbasis *website* untuk kelompok bermain dan taman kanak-kanak Permata Iman. Masalah yang melatarbelakangi penelitian tersebut yaitu proses penilaian evaluasi belajar siswa yang masih terbilang konvensional, guru yang menginginkan proses pembelajaran yang sejalan antara siswa belajar di sekolah dengan belajar di rumah. Penelitian tersebut memodelkan masalah yang didapat kedalam bentuk analisis kebutuhan sistem menggunakan *framework RUP (Rational Unified Process)*, menggunakan *Unified Modelling Language (UML)* sebagai bahasa pemodelan sistem berbasis *object oriented*. Keterkaitan penelitian tersebut ialah menggunakan orientasi yang sama dalam perancangan sistemnya, yaitu menggunakan *object oriented design*.

Penelitian ketiga yang dilakukan oleh Ependi pada tahun 2012 membahas tentang pendataan alumni secara lokal (*stay alone*) yang menyebabkan sulitnya pendistribusian data alumni terhadap perusahaan rekanan, sehingga melatarbelakangi pengembangan sistem.

Berdasarkan beberapa penelitian yang terkait, maka peneliti tertarik untuk melakukan penelitian tentang pembangunan sistem informasi manajemen akademik sekolah dasar menggunakan metode *Agile*, menggunakan *Unified Modelling Language* sebagai bahasa pemodelan sistemnya, dan mengimplementasikan sistem dalam bentuk *website*.

### 2.2 Sekolah Dasar

Sekolah merupakan tempat dimana pendidikan diselenggarakan melalui proses belajar dan mengajar. Berdasarkan pendekatan dari sudut pandang yang



sempit, pendidikan adalah segala kegiatan yang direncanakan dan dilakukan secara teratur pada instansi pendidikan sekolah (Suhartono, 2008). Pada dasarnya, sekolah dasar merupakan lembaga pendidikan yang menyelenggarakan program pendidikan selama enam tahun bagi anak berusia 6 sampai 12 tahun (Suharjo, 2006).

Sistem pendidikan nasional di Indonesia juga diatur dalam Undang-Undang Nomor 20, yang menyatakan bahwa pendidikan dasar merupakan prasyarat dasar untuk jenjang pendidikan yang lebih tinggi.

### 2.2.1 Sekolah Dasar Negeri Wates

Sekolah Dasar Negeri Wates merupakan instansi pendidikan yang terletak di Kecamatan Wates, Kabupaten Kediri, dan dimiliki oleh negara. Sekolah ini juga telah dinyatakan memenuhi 8 Standar Nasional Pendidikan Peraturan Menteri 19 Tahun 2005, yakni Standar Isi, Standar Proses, Standar Kompetensi Lulusan, Standar Pendidik dan Tenaga Kependidikan, Standar Sarana dan Prasarana, Standar Pembiayaan, dan Standar Penilaian Pendidikan.

SDN Wates memiliki visi, yaitu:

“Unggul dalam prestasi, beriman dan bertaqwa, berbudi pekerti luhur, serta berbudaya lingkungan.”

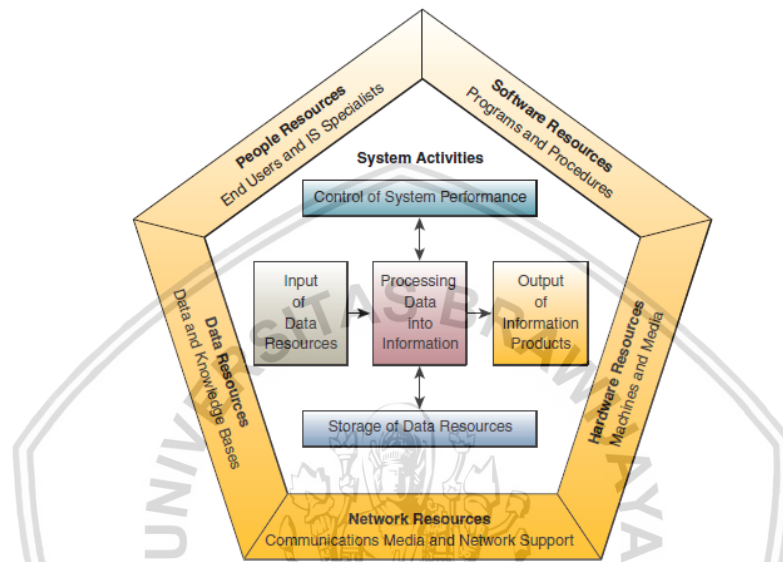
Serta memiliki Misi:

1. Terlaksananya suasana belajar yang PAIKEM, sehingga membangkitkan minat dan kreatifitas siswa dan guru dalam belajar dan berkarya.
2. Menghasilkan lulusan berpengetahuan luas dan mampu berfikir secara logis, mandiri dan kreatif.
3. Mendidik dan melatih peserta didik agar memiliki bekal melanjutkan ke jenjang SMP/MTs serta bekal kecakapan hidup.
4. Meningkatkan tersedianya sarana dan prasarana pembelajaran yang kondusif agar mampu mendukung peningkatan prestasi satuan pendidikan.
5. Terjalinnya hubungan yang harmonis dengan masyarakat (*stakeholder*) untuk mendukung upaya perlindungan dan pengolahan lingkungan hidup.

### 2.3 Sistem Informasi

Sistem didefinisikan sebagai seperangkat komponen yang saling terkait, memiliki batasan yang jelas, dan bekerjasama untuk mencapai suatu tujuan. Sedangkan sistem informasi ialah perpaduan dari manusia, perangkat lunak, perangkat keras, sumber data, jaringan komunikasi, kebijakan dan prosedur yang mengelola dan menyebarkan informasi dalam suatu organisasi. Sistem informasi digunakan manusia untuk berkomunikasi antara satu dengan lainnya menggunakan *hardware* sebagai perangkat fisik, *software* sebagai pemroses informasi dan prosedur, jaringan sebagai saluran komunikasi serta data yang tersimpan sebagai sumber data (O'Brien & Marakas, 2013).

Sistem informasi merupakan suatu sistem yang memperoleh data sebagai masukan (*input*), memproses data yang diterima sehingga menghasilkan *output* berupa informasi yang bernilai bagi pengguna, yang melibatkan berbagai komponen untuk mencapai tujuan. Model konseptual sistem informasi dimuat dalam Gambar 2.1. Semua sistem informasi melibatkan manusia, *software*, data, *hardware*, dan jaringan untuk melakukan aktivitas *input*, proses, *output*, penyimpanan dan pengendalian yang bertujuan untuk mengubah sumber data menjadi suatu informasi yang bernilai. (O'Brien & Marakas, 2013).



**Gambar 2.1 Sistem Informasi**

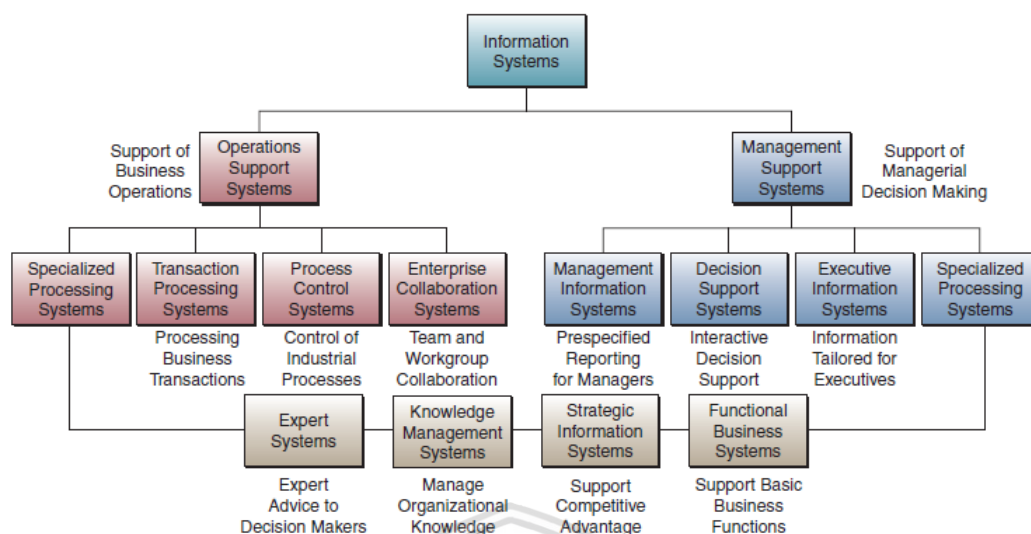
Sumber: O'Brien & Marakas (2013)

Penerapan sistem informasi dapat meningkatkan efisiensi suatu bisnis karena dengan adanya sistem informasi, informasi dapat diakses secara cepat, tepat, akurat, dan dapat dimanfaatkan untuk mencapai tujuan bisnis. Selain itu, kemudahan akses informasi juga dapat digunakan untuk membantu pengambilan keputusan.

### 2.3.1 Sistem Informasi Manajemen

Sistem Informasi Manajemen termasuk salah satu klasifikasi dari sistem informasi yang memberikan informasi berupa laporan dan *display* kepada para manajer dan profesional bisnis, dan merupakan bentuk sistem informasi yang paling umum dalam sebuah organisasi (O'Brien & Marakas, 2010). Contohnya, manajer bagian penjualan dapat menggunakan komputer yang terhubung dengan internet untuk mengakses laporan hasil penjualan produk secara *real-time* pada *web browser* dan dapat melakukan analisis penjualan harian untuk kebutuhan evaluasi. Gambar 2.2 mengklasifikasikan konseptual model dari sistem informasi berdasarkan tujuannya.





**Gambar 2.2 Klasifikasi Tipe Sistem Informasi**

Sumber: O'Brien & Marakas (2013)

## 2.4 Rekayasa Perangkat Lunak (*Software Engineering*)

Perangkat lunak (*Software*) adalah program komputer beserta dokumentasi yang menyertainya. Rekayasa perangkat lunak (*Software Engineering*) ialah disiplin ilmu yang berfokus pada segala aspek produksi perangkat lunak, mulai dari awal spesifikasi sistem hingga sistem jadi dan siap untuk digunakan. *Software Engineering* merupakan bagian dari rekayasa sistem. Segala aspek dalam produksi perangkat lunak bukan hanya berfokus pada teknik proses pengembangan perangkat lunak, namun juga termasuk aktivitas seperti manajemen proyek dan *tools* pengembangannya, metode, dan teori yang mendukung pengembangan perangkat lunak (Sommerville, 2011).

## 2.5 Agile Software Development

Bisnis sekarang beroperasi di lingkungan global yang berubah dengan cepat, seringkali tidak mungkin untuk mendapatkan seperangkat persyaratan perangkat lunak yang stabil. Proses pengembangan perangkat lunak harus dilakukan dengan cepat, untuk menghasilkan perangkat lunak yang berguna.

Metode *Agile* merupakan metode pengembangan inkremental dimana penambahannya kecil, dan biasanya sistem yang baru dirilis tersedia untuk klien setiap dua, atau tiga minggu. Metode ini melibatkan klien dalam proses pengembangan untuk mendapatkan umpan balik secara cepat mengenai perubahan persyaratan / kebutuhan sistem. Metode ini dalam dokumentasinya dilakukan secara minimal karena lebih menggunakan komunikasi secara informal dalam proses pengembangannya (Sommerville, 2011).

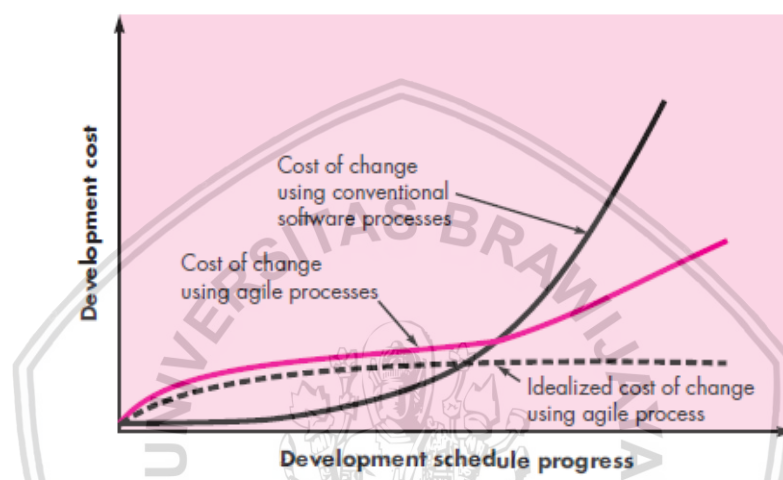
Prinsip dari *Agile Methods* yaitu:

1. *Customer Involvement* : Pelanggan harus secara dekat terlibat dalam proses pengembangan. Ini akan menyediakan prioritas baru dari kebutuhan sistem dan untuk mengevaluasi iterasi dari sistem.
2. *Incremental Delivery* : Software dikembangkan secara inkremental dengan pelanggan yang menspesifikasikan kebutuhan untuk diikuti dalam setiap perulangan (inkremen).
3. *People not Process* : Keterampilan tim pengembang harus diakui dan dieksplotasi. Anggota tim harus dibiarkan mengembangkan dengan cara mereka sendiri tanpa proses preskriptif.
4. *Embrace Change* : Mengharapkan kebutuhan sistem untuk terus berubah dan merancang sistem untuk mengakomodasi perubahan.
5. *Maintain Simplicity* : Fokus pada kesederhanaan baik dalam perangkat lunak yang dikembangkan maupun dalam proses pengembangannya. Sebaiknya mungkin aktif bekerja untuk mengeliminasi kompleksitas dari sistem.

Disamping itu *agilemanifesto.org* juga mengemukakan 12 prinsip *Manifesto for Agile Development*, yaitu:

1. Prioritas utama adalah untuk memuaskan pelanggan melalui *deliverable* perangkat lunak berkualitas yang cepat dan berkesinambungan.
2. Menyambut baik perubahan kebutuhan, bahkan jika terlambat dalam pengembangan. Perubahan dimanfaatkan sebagai keunggulan kompetitif pelanggan.
3. Serahkan perangkat lunak yang berfungsi sesering mungkin, dari beberapa minggu hingga beberapa bulan, dengan preferensi pada skala waktu yang lebih pendek.
4. Pebisnis dan pengembang harus bekerjasama setiap hari, selama proyek berlangsung.
5. Bangun proyek disekitar orang-orang yang termotivasi. Beri mereka lingkungan dan dukungan yang mereka butuhkan, dan yakin pekerjaan dapat terselesaikan.
6. Metode penyampaian informasi yang paling efisien dan efektif dalam tim pengembangan ialah percakapan secara tatap muka.
7. Perangkat lunak yang berfungsi adalah tolok ukur kemajuan yang utama.
8. Proses *agile* mempromosikan pengembangan berkelanjutan. Para pengembang, sponsor, dan pengguna harus bisa mempertahankan kecepatan konstan tanpa terbatas waktu.
9. Perhatian terus menerus terhadap desain yang baik dan keunggulan teknis akan meningkatkan kelincahan.
10. Sederhananya, "seni untuk memaksimalkan jumlah pekerjaan yang tidak dilakukan" sangat penting.
11. Arsitektur, kebutuhan, dan desain terbaik muncul dari kemampuan tim dalam mengorganisasikan.
12. Dalam interval reguler, tim merefleksikan bagaimana bekerja lebih efektif, kemudian menyesuaikan.

Berdasarkan beberapa prinsip yang telah dikemukakan, metode *agile* dikembangkan untuk digunakan oleh tim dengan skala kecil yang bisa bekerjasama di ruangan yang sama dan berkomunikasi secara informal. Metode ini telah banyak digunakan untuk pengembangan sistem berukuran kecil dan menengah. Pendekatan *agile* digunakan untuk pengembangan perangkat lunak, mempertimbangkan perancangan dan implementasi untuk dijadikan aktivitas utama dalam *software process*. Pendekatan ini menggabungkan aktivitas lain seperti *requirement elicitation and testing* kedalam desain dan implementasi (Sommerville, 2011). Gambar 2.3 menunjukkan perbandingan biaya perubahan dalam pengembangan antara pendekatan *agile* dengan pendekatan konvensional.



Gambar 2.3 Biaya Perubahan dalam Pengembangan

Sumber: Pressman (2010)

## 2.6 Proses Bisnis (*Business Process*)

Proses bisnis merupakan kumpulan aktivitas yang memerlukan satu atau lebih jenis *input* dan menjadikannya sebagai *output* yang bernilai bagi pengguna. Kumpulan aktivitas dalam proses bisnis dilakukan secara terkoordinasi untuk mewujudkan tujuan bisnis. Setiap proses bisnis diberlakukan oleh suatu organisasi, tetapi juga memungkinkan berinteraksi dengan proses bisnis yang dilakukan oleh organisasi lain (Weske, 2007).

Berikut ini adalah jenis aktivitas dalam proses bisnis (Weske, 2007):

1. *Manual Activity* (Aktivitas manual) merupakan aktivitas yang tidak didukung oleh sistem informasi. Contoh dari *manual activity* ialah mengirimkan parcel kepada rekan bisnis melalui perantara kurir.
2. *User Interaction Activity* (Aktivitas interaksi pengguna) merupakan aktivitas yang dilakukan oleh orang, dengan menggunakan sistem informasi. Agar pekerjaan yang dilakukan menjadi efektif, aktivitas ini perlu didukung antarmuka sistem yang baik. Contoh dari aktivitas ini yaitu memasukkan data pasien menggunakan sistem.

3. *System Activity* (Aktivitas system) merupakan aktivitas yang tidak melibatkan peranan dari pengguna. Aktivitas ini sepenuhnya dilakukan oleh sistem. Contoh dari aktivitas ini ialah menghitung jumlah total saldo rekening nasabah.

*Business process* dimodelkan melalui penggambaran diagram proses bisnis. Diagram proses bisnis merupakan diagram yang merepresentasikan *task-task* dan aktivitas dari suatu proses bisnis dan hubungan antara keduanya. Diagram proses bisnis digambarkan dengan menggunakan notasi grafik tertentu. Contoh notasi yang dapat digunakan untuk merepresentasikan model proses bisnis ialah *UML*: Diagram Aktivitas dan *Business Process Model and Notation (BPMN)* (Weske, 2007).



### 2.6.1 BPMN (*Business Process Model and Notation*)

*BPMN* merupakan notasi grafis yang berfungsi untuk memodelkan suatu *business process*. Tujuan utama dari *BPMN* ialah untuk memvisualisasikan notasi yang mudah dipahami oleh semua *stakeholder*, mulai dari *business analyst* yang membuat konsep awal dari keseluruhan proses, pengembang teknis yang berperan dalam pengimplementasian teknologi yang terlibat dalam proses tersebut, dan pelaku bisnis yang akan mengelola dan memonitor proses tersebut. (Weske, 2007).



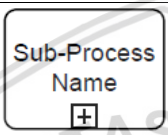
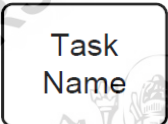


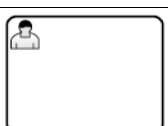
*BPMN* memiliki 4 kategori elemen dasar yaitu (Weske, 2007):

1. *Flow object*: Elemen utama yang menentukan perilaku proses bisnis. *Flow Object* terdiri dari *events*, aktivitas, dan *gateways*.
2. *Artefacts*: Digunakan untuk memvisualisasikan informasi tambahan tentang proses bisnis. *Artefacts* terdiri dari *groups*, *data object*, dan *text annotations*. Setiap *artefacts* dapat dikaitkan dengan *flow elements*.
3. *Connecting object*: Menghubungkan *swimlanes*, *artefacts*, atau *flow objects*. Elemen ini terdiri dari *message flows*, *associations*, *data associations*, dan *sequence flows*.
4. *Swimlanes*: Tersusun dari *pools* dan *lanes*. *Pools* menggambarkan organisasi yang berpartisipasi dalam interaksi beberapa proses bisnis, masing-masing *pools* dilakukan oleh satu organisasi. *Lanes* menggambarkan entitas organisasi, seperti departemen / divisi yang terdapat pada organisasi tersebut.

**Tabel 2.1 Simbol-Simbol Dalam BPMN**









Nama	Notasi	Deskripsi
<i>Start Event</i>		Merupakan notasi yang digunakan untuk memulai suatu proses
<i>End Events</i>		Merupakan notasi yang digunakan untuk mengakhiri suatu proses.

Tabel 2.1 Simbol-Simbol Dalam BPMN (Lanjutan)

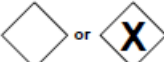







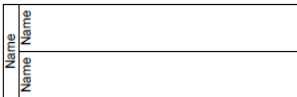
Nama	Notasi	Deskripsi
<i>Intermediate Events</i>		Menunjukkan suatu <i>event</i> (Kejadian) yang terdapat di antara awal dan akhir proses. Notasi ini hanya berpengaruh terhadap aliran proses
<i>Activity</i>		Notasi ini merupakan aktivitas / kegiatan yang dilakukan oleh perusahaan dalam suatu proses.
<i>Collapsed Sub-Process</i>		Notasi ini merupakan aktivitas yangmana didalam aktivitas tersebut masih terdapat proses yang terjadi.
<i>Task</i>		<i>Task</i> ialah bentuk atomik dari aktivitas. Task digunakan ketika pekerjaan dalam proses sudah tidak dapat dipecah lagi ke tingkat yang lebih detil.
<i>Service Task</i>		Merupakan <i>task</i> yang menggunakan otomatisasi layanan, dapat berupa aplikasi atau <i>web service</i> .
<i>Send Task</i>		<i>Task</i> yang digunakan untuk mengirim pesan kepada pihak eksternal dan selesai ketika pesan telah dikirim.
<i>Receive Task</i>		<i>Task</i> yang digunakan untuk menunggu dan menerima pesan yang dikirimkan oleh pihak eksternal. <i>Task</i> ini selesai ketika pesan telah diterima.
<i>User Task</i>		<i>Task</i> ini menunjukkan keterlibatan interaksi antara manusia dengan sistem / aplikasi.
<i>Manual Task</i>		<i>Task</i> ini menunjukkan bahwa <i>task</i> dilakukan tanpa melibatkan sistem.



Tabel 2.1 Simbol-Simbol Dalam BPMN (Lanjutan)





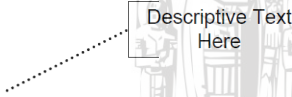
Nama	Notasi	Deskripsi
<i>Business Rule Task</i>		Menyediakan mekanisme proses untuk memberikan <i>input</i> ke <i>Business Rule Engine</i> dan menerima <i>output</i> yang diberikan <i>Business Rule Engine</i> .
<i>Script Task</i>		<i>Task</i> yang dilakukan oleh suatu mesin proses bisnis dengan menjalankan <i>script</i> .
<i>Sequence Flow</i>		Merepresentasikan alur dari <i>Flow Elements</i> dalam sebuah proses dan hanya memiliki satu <i>source</i> dan satu target.
<i>Conditional Flow</i>		Notasi ini digunakan ketika ada suatu kondisi, aliran akan diteruskan hanya jika kondisi bernilai benar / <i>true</i> .
<i>Default Flow</i>		Notasi ini digunakan hanya jika semua <i>conditional flow</i> bernilai <i>not true</i> / <i>false</i> pada saat <i>runtime</i> .
Message Flow		Notasi yang digunakan untuk menunjukkan aliran pengiriman pesan antara dua atau lebih partisipan.
<i>Association / Asosiasi</i>		Notasi yang digunakan untuk menghubungkan <i>artifact</i> dan informasi dengan <i>flow object</i> .
<i>Gateway</i>		Notasi ini menunjukkan percabangan (karna adanya kondisi) atau penggabungan ( <i>merge</i> ) <i>sequence flow</i> dalam sebuah proses.

Tabel 2.1 Simbol-Simbol Dalam BPMN (Lanjutan)

Nama	Notasi	Deskripsi
Gateway Control Types	<p>Exclusive </p> <p>Event-Based </p> <p>Parallel Event-Based </p> <p>Inclusive </p> <p>Complex </p> <p>Parallel </p>	<ul style="list-style-type: none"> <li>Keputusan (<i>Exclusive gateway</i>): digunakan untuk membuat jalur alternatif.</li> <li><i>Event-based gateway</i> : digunakan untuk pemilihan jalur berdasar <i>event</i>.</li> <li><i>Inclusive gateway</i>: digunakan untuk membuat jalur paralel dalam suatu aliran proses, dengan mengevaluasi semua kondisi.</li> <li><i>Complex gateway</i>: digunakan untuk mensinkronkan jalur kompleks.</li> <li><i>Parallel gateway</i>: digunakan untuk membuat aliran paralel.</li> </ul>
Pool		<i>Pool</i> merepresentasikan partisipan. Partisipan berupa suatu organisasi / perusahaan atau perwakilan peranan yang lebih umum (misalnya: produsen, konsumen).
Message		Merepresentasikan isi dari komunikasi yang dilakukan antara dua partisipan.
Lane		<i>Lane</i> merupakan bagian dalam proses atau <i>pool</i> yang memiliki kesamaan panjang dengan proses, baik secara vertikal maupun horizontal. Notasi ini digunakan untuk mengatur dan mengelompokkan aktivitas dalam sebuah <i>pool</i> . <i>Lane</i> terkadang digunakan untuk merepresentasikan peran internal perusahaan

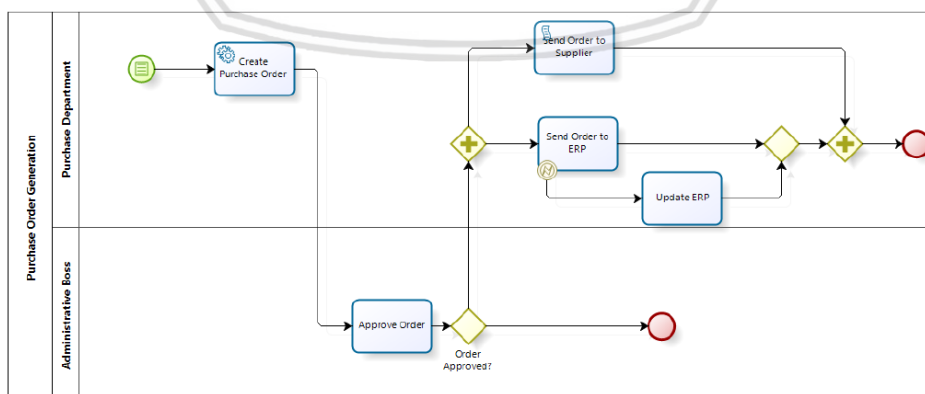


Tabel 2.1 Simbol-Simbol Dalam BPMN (Lanjutan)

Nama	Notasi	Deskripsi
<i>Data object</i>	<p>Data Object</p>  <p>Data Objec (Collection)</p>  <p>Data Input    Data Output</p> 	<ul style="list-style-type: none"> <li>• <i>Data Object</i>: merepresentasikan suatu informasi yang terdapat pada sebuah proses, dokumen, surat, <i>e-mail</i>.</li> <li>• <i>Collection</i>: merepresentasikan sekumpulan informasi.</li> <li>• <i>Data Input-Output</i>: <i>Data input</i> ialah masukan untuk keseluruhan proses, sedangkan <i>data ouput</i> ialah data hasil dari keseluruhan proses.</li> </ul>
<i>Group</i>		Notasi ini digunakan untuk mengelompokkan elemen-elemen grafis yang memiliki kategori yang sama. Nama kategori muncul sebagai label dari <i>group</i> .
<i>Text Annotation</i>		Notasi ini digunakan untuk memberikan informasi teks tambahan dalam sebuah proses.

Sumber: Object Management Group (OMG) (2011)

Contoh penggambaran diagram proses bisnis menggunakan *BPMN*:



Gambar 2.4 Contoh penggambaran diagram proses bisnis

Sumber: BPMN by example (2014)

## 2.7 Analisis Kebutuhan Sistem

Analisis kebutuhan sistem merupakan proses untuk membangun layanan atau produk yang dibutuhkan oleh pelanggan atau pengguna beserta batasan-batasan yang diberlakukan ketika produk atau layanan tersebut digunakan dan beroperasi, serta ketika produk atau layanan tersebut dibangun atau dikembangkan. Kebutuhan itu sendiri mengacu pada standar *IEEE* nomor: *IEEE-STD-1220-1998* menyatakan bahwa kebutuhan adalah sebuah pernyataan yang mengidentifikasi produk atau proses, baik secara operasional maupun fungsional, atau diperlukan batasan atau karakteristik dari desain, yang tidak ambigu, dapat diuji, atau terukur, dan agar produk dan prosesnya dapat diterima oleh konsumen atau sesuai dengan petunjuk penjaminan mutu secara internal.

Kebutuhan pengguna secara umum terbagi menjadi 2 kategori utama, yaitu kebutuhan fungsional dan kebutuhan non-fungsional. Kebutuhan fungsional merepresentasikan apa yang sistem lakukan, sedangkan kebutuhan non-fungsional merepresentasikan batasan apa saja yang harus dipenuhi oleh sistem dalam hal batasan kualitas dan batasan penggunaan (*usability*).

Secara umum, kebutuhan memiliki 3 tingkatan yakni:

- Kebutuhan normal: yaitu kebutuhan yang harus dipenuhi dan dinyatakan secara eksplisit oleh pengguna/stakeholder seperti fungsionalitas sistem dan performa sistem.
- Kebutuhan yang diharapkan (*expected*): yaitu kebutuhan yang tidak dinyatakan secara eksplisit namun menentukan kepuasan pelanggan, seperti: kemudahan interaksi dengan sistem, akurasi dan kebenaran proses.
- Kebutuhan yang mengejutkan (*exciting*): yaitu kebutuhan yang melebihi kebutuhan normal untuk dapat lebih memuaskan pelanggan, seperti fungsionalitas tambahan sistem lainnya yang lebih memuaskan pelanggan ketika menggunakan sistem dalam menyelesaikan masalah, mencapai tujuan pengguna, dengan lebih efisien.

Hasil dari proses analisis kebutuhan sistem diantaranya sebagai bentuk kesepakatan antara pengembang dengan pengguna akan kebutuhan yang harus dipenuhi, menyediakan dasar yang akurat bagi proses perancangan sistem, menjadi referensi bagi proses validasi untuk memastikan bahwa seluruh kebutuhan telah dispesifikasikan dengan benar, lengkap dan konsisten.

## 2.8 Perancangan Sistem

Perancangan merupakan proses untuk mendefinisikan sesuatu yang akan dikerjakan dengan teknik yang bervariasi, serta di dalamnya melibatkan deskripsi mengenai arsitektur serta detail komponen dan juga batasan yang dalam proses pengerjaannya. Menurut Pressman (2010), perancangan perangkat lunak (*software*) adalah proses dimana analisis diterjemahkan menjadi *blueprint* untuk membangun suatu perangkat lunak. *Blueprint* (rancangan) awalnya menggambarkan pandangan secara menyeluruh mengenai perangkat lunak, yaitu desain pada abstraksi tingkat tinggi, yang dapat langsung ditelusuri pada sistem

tertentu, objektif, terperinci, dan fungsionalitas. Namun perbaikan berikutnya mengarah pada representasi desain dengan tingkat abstraksi yang jauh lebih rendah.

Berikut ini merupakan prinsip dalam melakukan perancangan (Pressman, 2010):

- Hasil rancangan harus dapat dilacak dalam model persyaratan.
- Arsitektur sistem yang akan dibangun selalu dipertimbangkan.
- Rancangan dari data dan rancangan fungsi memiliki urgensi yang sama.
- Harus hati – hati dalam merancang antarmuka.
- Desain antarmuka pengguna harus sesuai dengan kebutuhan pengguna dan berfokus pada kemudahan pengguna.
- Komponen harus memiliki keterikatan yang rendah antara satu sama lain dan dengan lingkungan eksternal.
- Model rancangan harus mudah dimengerti.
- Rancangan harus dikembangkan secara iteratif.

Pada tahap ini memungkinkan tim pengembang untuk menilai dan meningkatkan kualitas dari hasil perancangan, sebelum lanjut ke tahap berikutnya. Model rancangan dinilai oleh tim pengembang sebagai upaya untuk menentukan apakah terdapat kesalahan, kelalaian, inkonsistensi? adakah alternatif yang lebih baik? Dan apakah model rancangan tersebut dapat dilaksanakan dalam jadwal, biaya, dan batasan yang telah ditetapkan.

*McGlaughlin* dalam *Pressman* (2010) mensyaratkan 3 karakteristik yang berguna sebagai panduan untuk evaluasi perancangan yang baik, yaitu:

1. Rancangan harus mengimplementasikan semua persyaratan eksplisit yang ada pada model persyaratan, dan harus memenuhi semua persyaratan implisit yang diinginkan oleh para *stakeholder*.
2. Rancangan harus dapat dibaca, dipahami untuk pengembang yang akan melakukan tahap implementasi dan pengujian, serta mereka yang akan mendukung perangkat lunak.
3. Rancangan harus menyediakan visualisasi secara lengkap dari perangkat lunak, fungsional, data, dan domain perilaku dari perspektif implementasi.

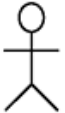


## **2.9 UML (Unified Modelling Language)**

*UML (Unified Modelling Language)* merupakan standar bahasa yang digunakan untuk kebutuhan pendefinisian rancangan dalam sebuah perangkat lunak. Bahasa ini digunakan untuk melakukan spesifikasi, visualisasi, konstruksi, dan pendokumentasian artefak dari perangkat lunak yang akan dikembangkan (Pressman, 2010). *UML* memiliki diagram-diagram yang digunakan untuk mendesain suatu sistem atau perangkat lunak, diantaranya yaitu: *Use Case Diagram*, *Activity Diagram*, *Sequence Diagram*, serta *Class Diagram*.

### 2.9.1 Use Case Diagram

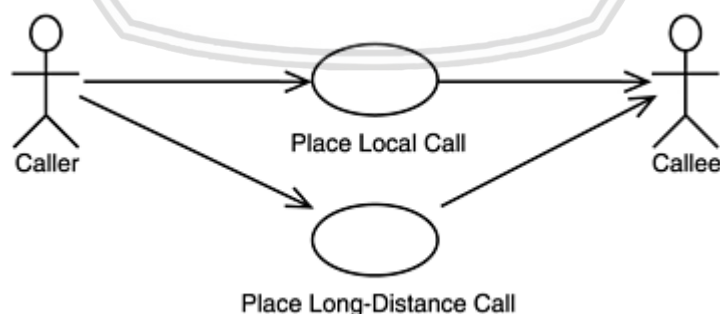
*Use Case* merupakan cara sederhana dan terbaik untuk mengekspresikan kebutuhan fungsional, maupun perilaku dari sistem (Bittner et.al, 2003). Sedangkan *use case specification* menggambarkan urutan kegiatan yang dilakukan oleh aktor dan sistem untuk mencapai suatu tujuan tertentu. *Use case specification* terdiri dari *use case name*, deskripsi singkat (*Brief Description*), Aliran Normal (*Basic flow*), Aliran Alternatif (*Alternate Flow*), *Special Requirements*, *Pre-Condition* dan *Post-Condition* (Pressman, 2010). Notasi yang digunakan pada diagram *use case* dapat dilihat pada Tabel 2.2

Tabel 2.2 Notasi Dalam Diagram *Use Case*

No	Simbol	Deskripsi
1	 An Actor	<b>Actors.</b> Mendefinisikan suatu peran yangmana pengguna dapat berinteraksi dengan sistem. Pengguna dapat berupa manusia ataupun sistem yang lain
2	 A Use Case	<b>Use Case.</b> Mendeskripsikan bagaimana aktor menggunakan sistem untuk mencapai tujuan tertentu dan apa yang sistem lakukan untuk aktor untuk mendapatkan tujuannya.
3		<b>Communicates association.</b> Aktor dan <i>use case</i> yang berinteraksi akan terhubung melalui symbol ini.  Simbol asosiasi dengan panah mengindikasikan elemen mana yang memulai interaksi.

Sumber: Kurt Bittner & Ian Spence (2003)

Contoh penggambaran diagram *Use Case*:




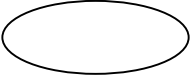

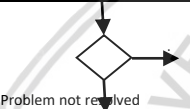
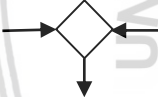



Gambar 2.5 Contoh penggambaran diagram *Use Case*

Sumber: Kurt Bittner & Ian Spence (2003)

### 2.9.2 Activity Diagram

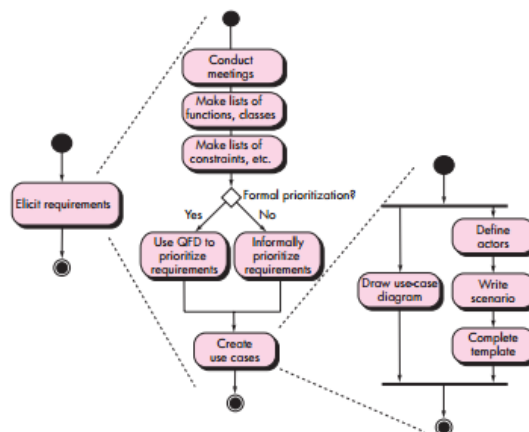
*Activity Diagram* / diagram aktivitas merupakan salah satu dari diagram *UML* yang digunakan untuk memodelkan aspek dinamis (*Scenario-based elements*) dari sistem. Representasi grafis berupa aliran dari interaksi berdasarkan scenario spesifik (Pressman 2010). Notasi yang digunakan pada diagram *activity* dapat dilihat pada Tabel 2.3

**Tabel 2.3 Notasi Dalam Diagram Activity**

No	Simbol	Deskripsi
1		<b>Initial node.</b> Mendefinisikan titik mulainya suatu proses.
2		<b>Actions.</b> Mendefinisikan notasi yang memvisualisasikan langkah-langkah yang terjadi.
3		<b>Flow.</b> Berupa gambar panah dalam diagram yang mengindikasikan alur antar <i>actions</i> .
4		<b>Decisions.</b> Merupakan kondisi tertentu yang menentukan alur keluar.
5		<b>Merge.</b> Merupakan penggabungan alur yang sebelumnya terpisah oleh <i>decisions</i> .
6		<b>Fork.</b> Merupakan <i>bar</i> hitam untuk percabangan aktivitas yang secara bersamaan.
7		<b>Join.</b> Untuk menyatukan kembali alur yang terpisah oleh <i>fork</i> .
8		<b>Activity Final.</b> Merupakan titik akhir proses.

Sumber: Booch et. Al. (2005)

Contoh penggambaran diagram *Activity*:



Gambar 2.6 Contoh penggambaran diagram *Activity*

Sumber: UML 2.0

### 2.9.3 Sequence Diagram

Diagram *Sequence* merupakan diagram interaksi yang menekankan pada saat permintaan pesan yang menunjukkan serangkaian peranan dan pesan yang dikirim dan diterima oleh bagian yang ada pada peranan tersebut. Notasi pesan untuk panggilan terhadap suatu operasi dapat dituliskan dalam bahasa *UML* atau dalam bahasa pemrograman tertentu. *Sequence diagram* digunakan untuk memvisualisasikan *dynamic view* dari sistem (Booch, et al., 2005). Notasi yang terdapat pada diagram *Sequence* ditunjukkan pada Tabel 2.4

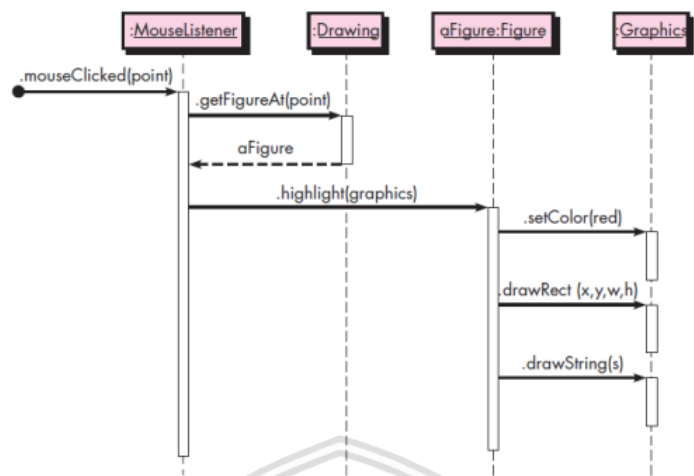
Tabel 2.4 Notasi Dalam Diagram *Sequence*

No	Simbol	Deskripsi
1		<b>Entity.</b> Mendefinisikan suatu entitas yang mempunyai atribut untuk keperluan penyimpanan data.
2		<b>Boundary.</b> Mendefinisikan suatu <i>interface</i> yang menghubungkan pengguna dengan sistem.
3		<b>Control.</b> Mendefinisikan kontrol atas aktivitas yang dilakukan oleh sebuah kegiatan.
4		<b>Message.</b> Mendefinisikan pengiriman pesan.
5		<b>Return Values.</b> Mendefinisikan hasil dari pengiriman pesan.
6		<b>Lifelines.</b> Mengindikasikan urutan dari kehidupan.
7		<b>Bar.</b> Menunjukkan periode waktu ketika aktif dalam interaksi.

Sumber: Booch, et.al (2005)



Contoh penggambaran diagram *Sequence*:



Gambar 2.7 Contoh diagram *Sequence*

Sumber: Pressman (2010)

2.9.4 Class Diagram


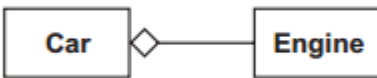

*Class diagram* atau diagram kelas digunakan untuk memvisualisasikan tampilan statis dari sebuah sistem. Selain penting untuk menentukan, memvisualisasikan, dan mendokumentasikan model struktural, diagram kelas juga penting untuk menyusun sistem yang dapat dilakukan melalui teknik *forward* maupun *reverse*. *Class diagram* secara umum memuat tiga elemen, yaitu *class*, *interfaces*, dan hubungan dependensi, generalisasi, serta asosiasi (Booch, et al., 2005). Tabel 2.5 menjelaskan tentang notasi yang digunakan dalam pembuatan *class diagram* pada umumnya.

Tabel 2.5 Notasi Dalam Diagram *Class*

No	Simbol	Deskripsi
1		<p><b>Attribute.</b> Mendefinisikan kumpulan data yang dimiliki oleh suatu objek.</p> <p><b>Behavior.</b> Merupakan kumpulan dari perilaku yang dapat dilakukan oleh objek dan terkait dengan fungsi-fungsi yang bertindak pada data objek (atribut). Perilaku objek merujuk kepada operasi, metode, atau fungsi.</p>
2		<p><b>Inheritance.</b> Mengindikasikan bahwa suatu kelas merupakan turunan dari kelas yang lain.</p>

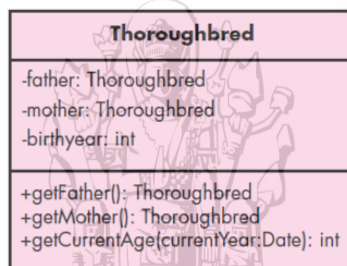


Tabel 2.5 Notasi Dalam Diagram *Class* (Lanjutan)

No	Simbol	Deskripsi
3		<b>Association.</b> Mengindikasikan bahwa objek dari satu kelas berhubungan dengan kelas yang lain.
4		<b>Aggregation.</b> Menunjukkan bahwa contoh objek dari satu kelas terdiri dari contoh objek dari kelas yang lain.
5		<b>Composition.</b> Menunjukkan asosiasi dimana satu kelas bertanggung jawab atas pembuatan dan kerusakan bagian-bagian dalam kelas lainnya. Jika satu kelas rusak, maka kelas lain juga rusak.

Sumber: Booch, et.al (2005)

Contoh penggambaran diagram *Class*:



Gambar 2.8 Contoh diagram *Class*

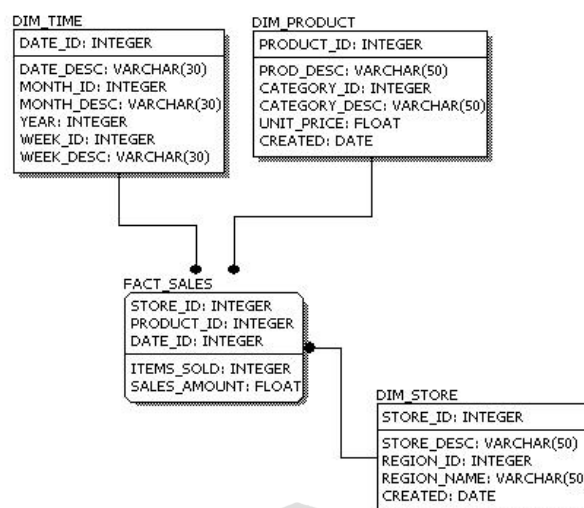
Sumber: Pressman (2010)

### 2.9.5 Physical Data Model

*Physical data model* merepresentasikan bagaimana model akan dibangun di *database*. Sebuah *physical data model* menampilkan struktur tabel yang terdiri dari nama kolom, tipe data kolom, *constraints* kolom, *primary key*, *foreign key*, dan relasi antara tabel (1keydata, 2017). Terdapat beberapa tahap dalam pembuatan *physical data model*, yakni

1. Merubah entitas menjadi tabel.
2. Merubah relasi menjadi *foreign key*.
3. Merubah atribut menjadi kolom.
4. Memodifikasi *physical data model* sesuai dengan kebutuhan.

Pada Gambar 2.9 dapat dilihat contoh *physical data model*.



**Gambar 2.9 Contoh *physical data model***

Sumber: 1keydata(2017)

## 2.10 Implementasi Sistem

Implementasi sistem merupakan salah satu tahapan dari siklus pengembangan sistem, dimana desain / rancangan sistem akan diterjemahkan kedalam bentuk bahasa pemrograman tertentu, sehingga menjadikan sistem yang berfungsi secara utuh.

### 2.10.1 *Object Oriented Programming (OOP)*

*Object Oriented Programming* merupakan salah satu paradigma dalam pembangunan sistem yang memandang segala sesuatu merupakan objek dan mempunyai atribut yang melekat, serta memiliki *behavior* (Shalahudin & S, 2015). Keuntungan mengembangkan sistem dengan menggunakan *object oriented programming* yaitu:

1. Meningkatkan produktivitas, karena objek yang dibuat dapat dipakai ulang (*reusable*).
2. Meningkatkan kecepatan dalam mengembangkan sistem karena telah dirancang dengan baik dan benar sehingga kesalahan dalam implementasi dapat diminimalisir.
3. Mudah dalam pemeliharaan karena menggunakan objek.
4. Adanya konsistensi dengan menggunakan konsep pewarisan (*inheritance*).
5. Meningkatkan kualitas dari perangkat lunak karena pengembangan didasari atas kejadian nyata, sehingga kebutuhan pengguna terpenuhi.

### 2.10.2 *Model-View-Controller (MVC) Design Pattern*

*Model-View-Controller (MVC)* adalah pola arsitektur yang membagi aplikasi ke dalam 3 komponen yang saling berinteraksi satu dengan yang lainnya (Sommerville, 2011). Masing-masing komponen pada pola arsitektur *MVC* memiliki peran yang berbeda.

Sommerville menjelaskan, dalam konteks aplikasi berbasis web, peran komponen model adalah objek yang menyediakan fungsi yang berisi logika bisnis atau fungsi untuk mengakses data yang akan dibutuhkan oleh komponen *view* untuk ditampilkan kepada pengguna.

Komponen *view* memiliki peran sebagai objek yang menampilkan informasi kepada pengguna. Informasi yang ditampilkan melalui *view* dapat berubah sesuai dengan perubahan data dan keadaan objek *model*. Penggunaan *MVC* memungkinkan implementasi beberapa objek *view* untuk sebuah objek *model* yang sama untuk memenuhi kebutuhan variasi presentasi informasi.

Selanjutnya, komponen *controller* memiliki peran untuk menerima dan mengolah interaksi pengguna yang diterima oleh komponen *view*. Selain itu, komponen *controller* juga dapat digunakan sebagai objek untuk memvalidasi data yang didapatkan dari interaksi pengguna terhadap sistem melalui komponen *view*. Komponen *controller* juga dapat berinteraksi dengan komponen *model* untuk mengirimkan permintaan pembaruan data.

### 2.10.3 Website

*Website* (situs web) merupakan kumpulan dari halaman web yang menampilkan informasi berupa teks, gambar, animasi, suara, video, atau kombinasi dari semuanya baik yang bersifat statis maupun dinamis yang membentuk rangkaian komponen saling terkait dimana masing-masing dihubungkan dengan *hyperlink*. Sebuah *website* dikatakan statis apabila isi dari informasinya tetap dan hanya bisa dirubah oleh pemilik *website*. Sedangkan dikatakan dinamis apabila konten dari *website* dapat diubah secara interaktif oleh pengguna maupun pemiliknya.

### 2.11 Pengujian Sistem

Pengujian sistem merupakan proses menjalankan suatu program atau sistem yang bertujuan untuk menemukan kesalahan dari sistem yang telah dikembangkan, namun bukan untuk menunjukkan bahwa program tersebut salah. Pengujian yang baik yaitu pengujian yang mempunyai kemungkinan tinggi untuk menemukan kesalahan yang belum pernah ditemukan sebelumnya, dan memiliki probabilitas yang tinggi untuk menemukan suatu kesalahan (Pressman, 2010). Menurut Pressman, terdapat beberapa prinsip dalam melakukan pengujian system, yakni:

1. Pengujian fokus pada usaha untuk menemukan kesalahan pada modul yang terintegrasi, dan hingga pada keseluruhan sistem.
2. Pengujian yang dilakukan memiliki kemungkinan tinggi dalam menemukan kesalahan.
3. Pengujian yang dilakukan harus sesuai dengan persyaratan pelanggan.
4. Para ahli sistem harus mengetahui spesifikasi dari produk sistem / perangkat lunak yang diinginkan pelanggan.
5. Perekayasa perangkat lunak bukan orang yang paling tepat untuk melakukan semua pengujian bagi sistem / perangkat lunak.

### 2.11.1 Black-Box Testing

*Black-Box Testing* adalah teknik pengujian perangkat lunak yang berfokus pada pengecekan kebutuhan fungsional sistem. Pengujian ini dilakukan untuk mengetahui kesalahan / *error* yang terdapat pada perangkat lunak. *Error* yang dimaksud yakni antara lain kesalahan atau hilangnya fungsi, kesalahan antarmuka, kesalahan pada basis data, kesalahan pada kinerja, serta kesalahan inisialisasi dan terminasi (Pressman, 2010). Pengujian ini memiliki beberapa keunggulan sebagai berikut:

1. Pengujian dilakukan berdasar sudut pandang pengguna dan akan membantu dalam mengungkap ketidaksesuaian dalam spesifikasi perangkat lunak.
2. *Tester* tidak perlu mengetahui bahasa pemrograman, atau bagaimana mengimplementasikan perangkat lunak.
3. Pengujian dapat dilakukan oleh badan independen dari pengembang, dan memungkinkan memiliki perspektif objektif.
4. *Test case* dapat langsung dirancang setelah spesifikasinya lengkap.

#### 2.11.1.1 Validation Testing

*Validation Testing* merupakan teknik pengujian dalam lingkup metode *Black Box* yang mana kesesuaian di uji. Pengujian ini dilakukan untuk mengetahui apakah sistem yang dibangun berjalan sesuai dengan kebutuhan yang telah didefinisikan atau tidak. Pengujian ini termasuk dalam *black-box testing*, karena pengujian dilakukan tanpa menunjukkan proses berjalannya sebuah fungsi yang sebenarnya, tetapi pengujian validasi melihat hasil yang dikeluarkan apakah sesuai dengan kebutuhan atau tidak, dan lebih ditekankan untuk menemukan konformitas antara kinerja sistem dengan daftar spesifikasi kebutuhan (Indriati, 2010).

#### 2.11.2 User Acceptance Testing (UAT)

*User Acceptance Testing (UAT)* adalah pengujian yang menangani requirement, business process, dan kebutuhan pengguna. Pengujian ini dilakukan untuk mengetahui apakah suatu sistem yang dibangun telah memenuhi kriteria untuk diterima oleh pengguna ataukah tidak (Hambling, 2013).

*User Acceptance Testing* biasanya dilakukan sebelum produk dirilis dan dilakukan setelah pengujian sistem. UAT dilakukan oleh pengguna / stakeholder yang terlibat dengan sistem yang dibangun.

### 2.12 Skala Likert

Skala *Likert* merupakan salah satu cara yang berguna dalam mempelajari hubungan antara sikap dan atau persepsi individu diperiksa dan dianalisis (McLeod, et al, 2011). Pada implementasinya, skala ini digunakan dengan cara meminta beberapa responden untuk memberikan opini mengenai pertanyaan yang diajukan. Responden akan menilai sejauh mana mereka setuju ataupun tidak setuju mengenai solusi dari permasalahan yang ditentukan. Proses perhitungan pada skala *likert* diawali dengan memberikan bobot nilai pada setiap jawaban.

Tabel 2.6 berisi rincian bobot nilai jawaban, sedangkan tabel 2.7 berisi persentase untuk setiap jawaban.

**Tabel 2.6 Rincian Bobot Nilai**

A = Sangat Setuju	5
B = Setuju	4
C = Netral	3
D = Kurang	2
E = Sangat Kurang	1

Sumber: Choizes (2017)

**Tabel 2.7 Persentase Nilai**

80 sampai 100%	Sangat Setuju
60 sampai 79,99%	Setuju
40 sampai 59,99%	Netral
20 sampai 39,99%	Tidak Setuju
0 sampai 19,99%	Sangat Kurang Setuju

Sumber: Choizes (2017)

Setelah bobot dan persentase nilai ditentukan, selanjutnya menghitung jumlah jawaban, dikalikan dengan bobot nilai. Berikut merupakan perhitungannya :

1. Jawaban Sangat Setuju (A) =  $nx5$
2. Jawaban Setuju (B) =  $nx4$
3. Jawaban Netral (C) =  $nx3$
4. Jawaban Tidak Setuju (D) =  $nx2$
5. Jawaban Sangat Kurang Setuju (E) =  $nx1$

$$TotalNilai = (nx5) + (nx4) + (nx3) + (nx2) + (nx1) \quad (2.1)$$

$n$  merupakan jumlah responden yang menjawab. Untuk melakukan perhitungan selanjutnya harus diketahui terlebih dahulu nilai terendah ( $Y$ ) dan nilai tertinggi ( $X$ ). Rumus yang digunakan dapat dilihat pada Rumus 2.2 dan 2.3, dimana  $N1$  merupakan nilai tertinggi *likert*,  $N2$  merupakan nilai terendah *likert*,  $n$  merupakan jumlah responden, dan  $U$  merupakan jumlah uji kasus.

$$Y = N1 \times nxU \quad (2.2)$$

$$X = N2 \times nxU \quad (2.3)$$

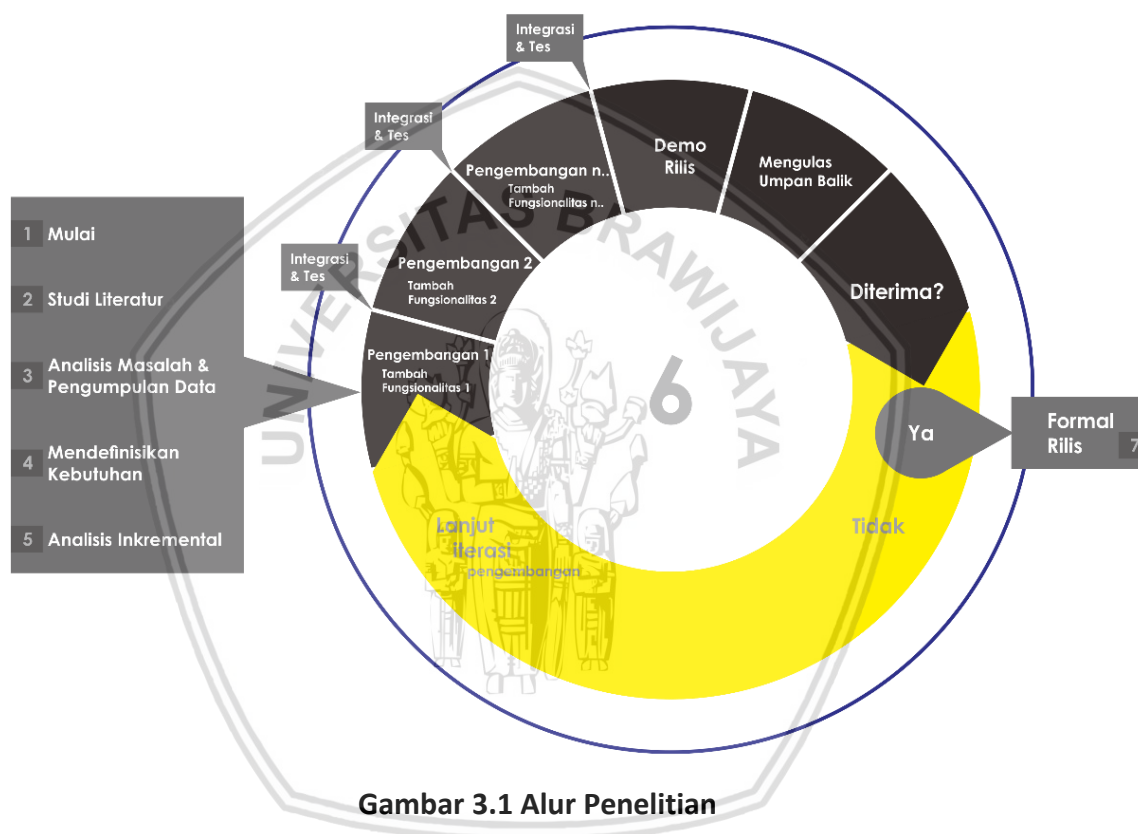
Sedangkan untuk menghitung hasil dari  $UAT$  dengan skala *likert* terdapat pada rumus 2.4.

$$RumusIndex\% = (TotalNilai / Y) \times 100 \quad (2.4)$$



## BAB 3 METODOLOGI PENELITIAN

Penelitian ini merupakan penelitian implementatif-konstruksi, dimana produk yang dihasilkan berupa sistem informasi yang siap untuk digunakan. Pendekatan implementatif yang digunakan dalam penelitian ini menggunakan pendekatan berorientasi objek. Langkah - langkah yang akan dilakukan dalam penelitian ini ialah studi literatur, analisis masalah dan pengumpulan data, mendefinisikan kebutuhan, analisis inkremental, pengembangan pertama sampai ke-n, integrasi dan testing, demo rilis, hingga formal rilis. Gambar 3.1 menyajikan alur penelitian yang akan dilakukan dalam penelitian ini.



### 3.1 Studi Literatur

Studi literatur merupakan tahapan mencari, mempelajari dan menyusun dasar teori dan referensi yang mendukung pembangunan sistem informasi. Literatur yang digunakan berasal dari paper / jurnal, buku, atau penelitian yang sebelumnya telah dilakukan oleh peneliti lain. Dalam penelitian ini dijelaskan teori mengenai pembangunan sistem informasi, *agile methodology*, analisis kebutuhan, perancangan berorientasi objek, website, pengujian menggunakan metode *black box – validation testing* dan *user acceptance testing*.

### 3.2 Analisis Masalah & Pengumpulan Data

Setelah melakukan studi literatur, tahapan selanjutnya ialah melakukan analisis masalah yang terdapat pada bagian akademik. Pada tahap ini dilakukan



wawancara kepada pihak sekolah untuk mengumpulkan masalah, dan informasi yang kemudian dilakukan analisis masalah mana yang dapat dijadikan sebagai latar belakang penelitian, dan bisa diselesaikan dengan menggunakan metode serta literatur yang dipakai. Hasil wawancara berupa data yang dikumpulkan sebagai dasar untuk mendefinisikan kebutuhan sistem informasi yang akan dibangun.

### 3.3 Mendefinisikan Kebutuhan

Pada tahap ini identifikasi dilakukan mulai dari pemberlakuan aturan penomoran untuk artefak pengembangan sistem, kemudian memodelkan proses bisnis saat ini dan proses bisnis usulan menggunakan *Business Process Model Notation (BPMN)*. Hasil aktivitas dari proses bisnis usulan dikodekan dan diidentifikasi aktor yang terlibat dalam sistem, kemudian mengidentifikasi kebutuhan fungsional dan non-fungsional dari proses bisnis usulan. Dengan menggunakan konsep OOAD (Object Analysis And Design), kebutuhan fungsional dimodelkan kedalam *Use Case Diagram* untuk setiap fitur yang akan dibangun serta membuat skenario dari setiap *Use Case*. Tujuan dari tahap ini ialah menspesifikasikan kebutuhan sistem informasi manajemen akademik sekolah, serta hasil luarannya berupa dokumen spesifikasi kebutuhan.

### 3.4 Analisis Inkremental

Analisis inkremental dilakukan untuk menentukan inkremen berdasarkan prioritas kebutuhan yang telah didefinisikan pada tahap analisis kebutuhan. Hasil dari analisis inkremental digunakan sebagai acuan untuk urutan dari pengembangan.

### 3.5 Pengembangan Pertama Sampai ke-N

Tahap ini merupakan tahap dimana perancangan dan implementasi dilakukan. Perancangan meliputi pembuatan diagram aktivitas, diagram sekuen, diagram kelas, pemodelan data, perancangan algoritma, serta perancangan antarmuka pengguna. Implementasi meliputi implementasi diagram kelas, implementasi fungsi / algoritma, implementasi DDL (*Data Definition Language*), serta implementasi halaman antarmuka pengguna. Perancangan dan Implementasi keseluruhan diselesaikan secara bertahap pada setiap fiturnya (fungsionalitas). Tujuan dari tahap ini ialah memodelkan rancangan sistem informasi manajemen akademik sekolah dan mengimplementasikan rancangan kedalam bahasa pemrograman web, sehingga hasil luarannya berupa dokumen perancangan dan sistem informasi berbasis *website*.

### 3.6 Integrasi & Testing

Integrasi dan *Testing* dilakukan pada setiap inkremen pengembangan sistem informasi, dilakukan secara bertahap pada setiap fitur yang dikembangkan. *Testing* dilakukan menggunakan metode *black box – validation testing* pada fitur yang dibangun. Tujuan dari tahap ini ialah menguji fitur pada sistem informasi

manajemen akademik sekolah, sehingga hasil luarannya berupa dokumen pengujian.

### **3.7 Demo Rilis**

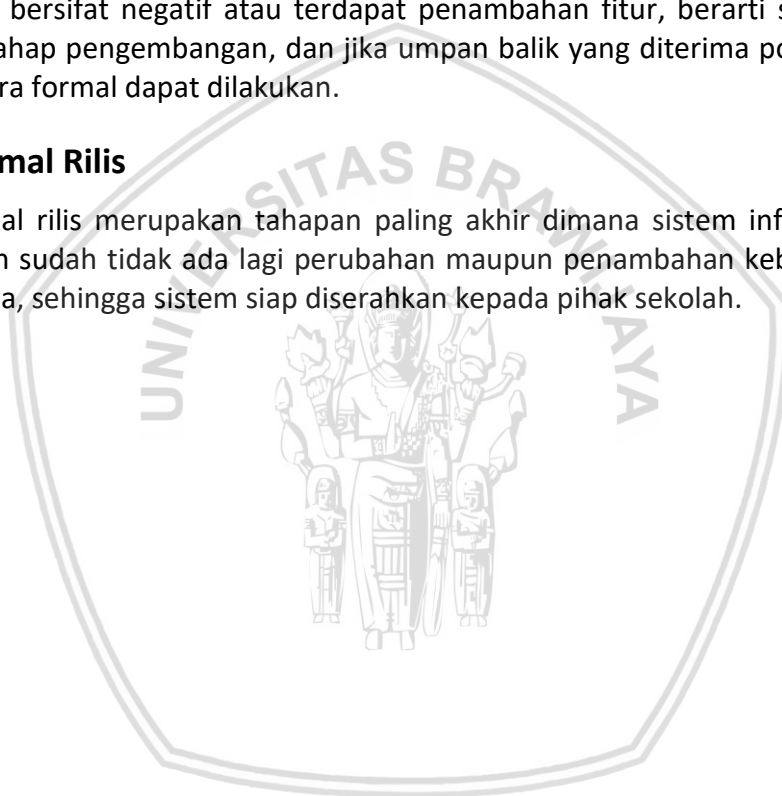
Setelah sistem informasi teruji dengan baik dan semua fungsionalitas telah tercukupi, maka demo rilis akan dilakukan. Demo rilis diberlakukan kepada pihak sekolah dengan cara menjalankan sistem informasi yang telah dibuat.

### **3.8 Mengulas Umpan Balik**

Setelah demo rilis dilakukan, umpan balik atas penggunaan sistem informasi yang dibangun akan dijadikan dasar untuk rilis secara resmi, jika umpan balik yang diterima bersifat negatif atau terdapat penambahan fitur, berarti sistem harus masuk tahap pengembangan, dan jika umpan balik yang diterima positif, berarti rilis secara formal dapat dilakukan.

### **3.9 Formal Rilis**

Formal rilis merupakan tahapan paling akhir dimana sistem informasi yang dibangun sudah tidak ada lagi perubahan maupun penambahan kebutuhan dari pengguna, sehingga sistem siap diserahkan kepada pihak sekolah.

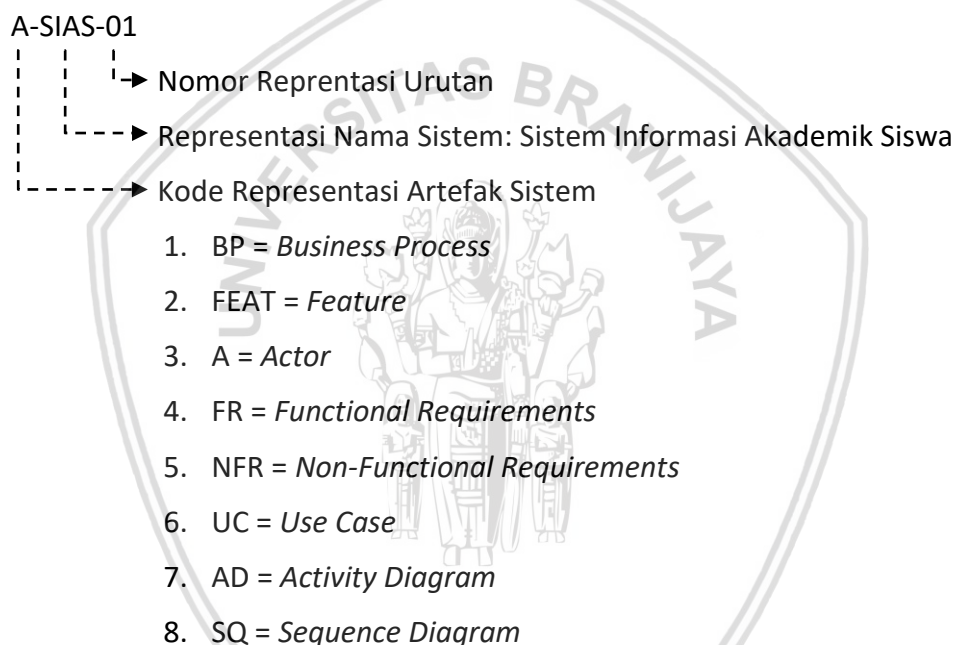


## BAB 4 ANALISIS KEBUTUHAN

Dalam pembangunan sistem informasi manajemen akademik yang akan dilakukan, proses bisnis dimodelkan untuk mengetahui bagaimana akademik berjalan sebelum dan sesudah adanya sistem informasi manajemen, analisis kebutuhan dilakukan mulai aturan penomoran, identifikasi aktor, menspesifikasikan kebutuhan fungsional maupun non fungsional sistem, menggambarkan kebutuhan kedalam *use case diagram*, hingga menuliskan *use case scenario*.

### 4.1 Aturan Penomoran

Aturan penomoran digunakan untuk memudahkan kodifikasi dan *traceability*. Berikut aturan penomoran yang digunakan, serta realisasi aturan pada tabel 4.1



**Tabel 4.1 Realisasi Aturan Penomoran**

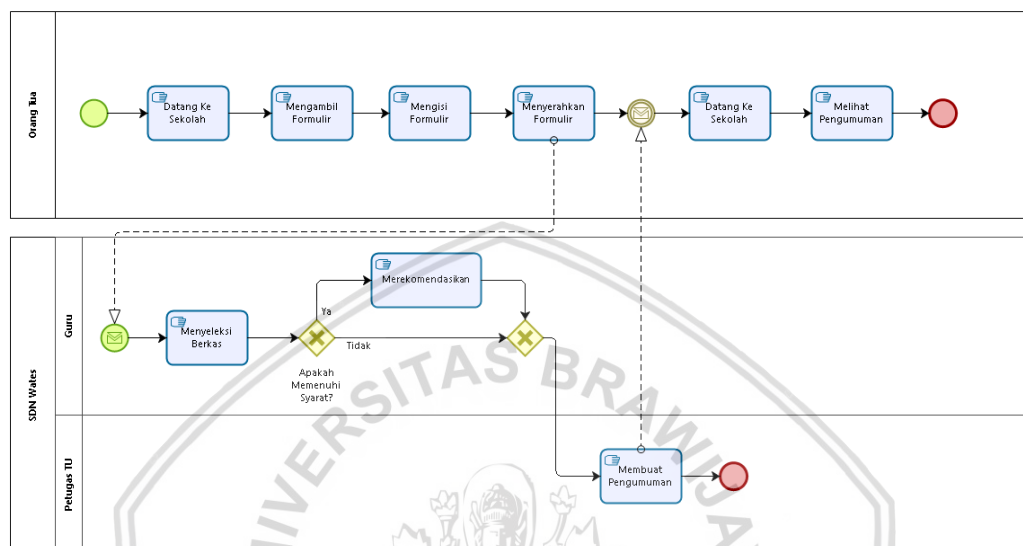
Artefak	Aturan Penomoran	Contoh Realisasi
<i>Business Process</i>	BP-SIAS-x	BP-SIAS-1
Identifikasi Fitur	FEAT-SIAS-x	FEAT-SIAS-1
Identifikasi Aktor	A-SIAS-x	A-SIAS-1
Kebutuhan Fungsional	F-SIAS-x	F-SIAS-1
Kebutuhan Non-Fungsional	NF-SIAS-x	NF-SIAS-1
<i>Use Case</i>	UC-SIAS-x	UC-SIAS-1
<i>Activity Diagram</i>	AD-SIAS-x	AD-SIAS-1
<i>Sequence Diagram</i>	SQ-SIAS-x	SQ-SIAS-1

## 4.2 Pemodelan Proses Bisnis

Proses bisnis dimodelkan dengan menggunakan *BPMN (Business Process Modeling Notation)* terhadap proses bisnis saat ini, dan proses bisnis usulan.

### 4.2.1 Registrasi

Pada proses Registrasi untuk saat ini, dapat dilihat pada gambar 4.1 berikut:

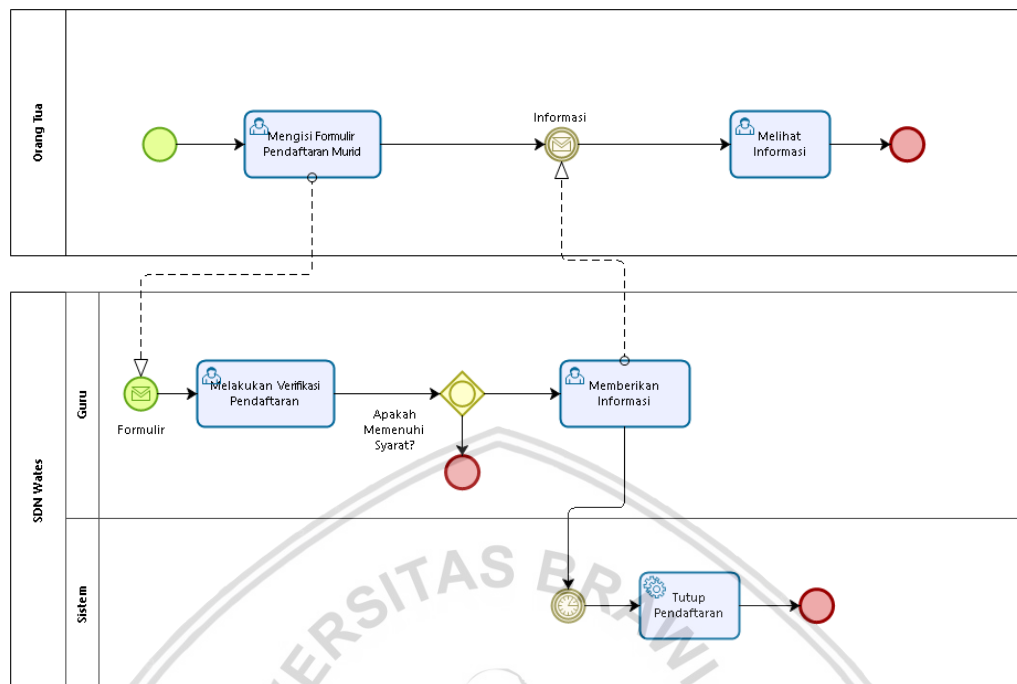


**Gambar 4.1 Proses Bisnis Registrasi Saat Ini**

Uraian proses bisnis registrasi saat ini yang terdapat pada gambar 4.1 ialah sebagai berikut:

1. Proses bisnis registrasi dimulai dengan aktivitas orang tua datang ke sekolah.
2. Kemudian orang tua mengambil formulir pendaftaran murid.
3. Orang tua mengisi formulir pendaftaran dengan lengkap, setelah itu menyerahkan formulir kepada guru.
4. Guru menerima formulir pendaftaran murid dari orang tua.
5. Kemudian guru menyeleksi berkas, apakah memenuhi syarat atau tidak? Jika memenuhi syarat, maka calon murid direkomendasikan untuk dijadikan murid, jika tidak memenuhi syarat, maka tidak direkomendasikan.
6. Petugas TU membuat pengumuman daftar calon murid yang diterima, yang dipasang di papan pengumuman pada tanggal yang telah ditentukan.
7. Orang tua datang ke sekolah untuk melihat pengumuman penerimaan murid.

Sedangkan proses Registrasi usulan, dapat dilihat pada gambar 4.2 berikut:



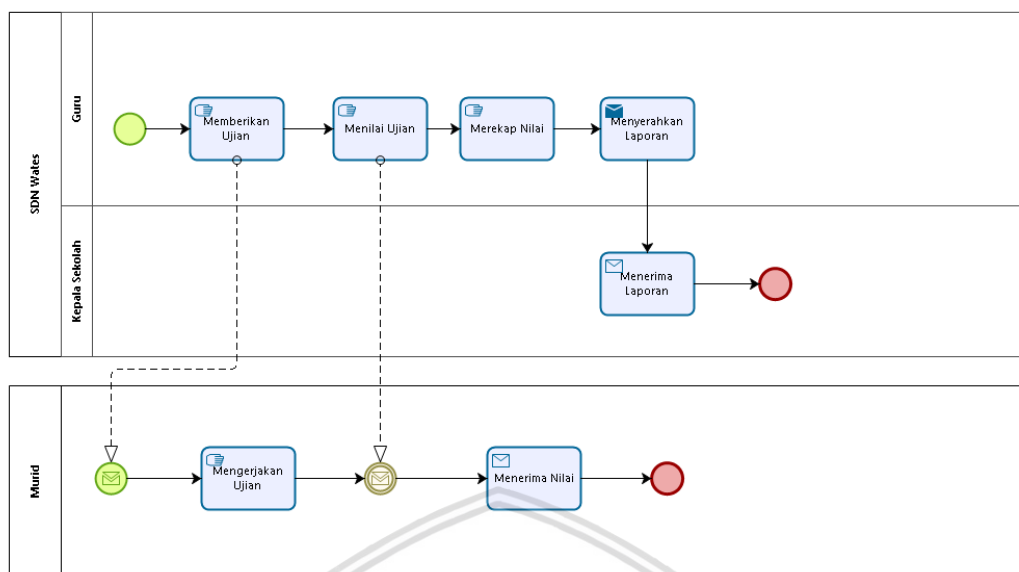
**Gambar 4.2 Proses Bisnis Registrasi Usulan**

Uraian proses bisnis registrasi saat usulan yang terdapat pada gambar 4.2 ialah sebagai berikut:

1. Proses bisnis dimulai dengan aktivitas orang tua mengisi formulir pendaftaran murid, pada tanggal yang telah ditentukan.
2. Guru kemudian melakukan verifikasi pendaftaran murid yang diisi orang tua. Jika datanya memenuhi kriteria, maka guru merekomendasikan untuk diterima, sedangkan jika datanya tidak memenuhi kriteria, maka tidak akan direkomendasikan. Guru kemudian memberikan informasi ke orang tua melalui email.
3. Orang tua melihat informasi pendaftaran melalui email.
4. Sistem kemudian memberlakukan tutup pendaftaran pada jadwal yang telah ditentukan.

#### 4.2.2 Mengevaluasi Murid

Proses evaluasi murid untuk saat ini, dapat dilihat pada gambar 4.3 berikut:



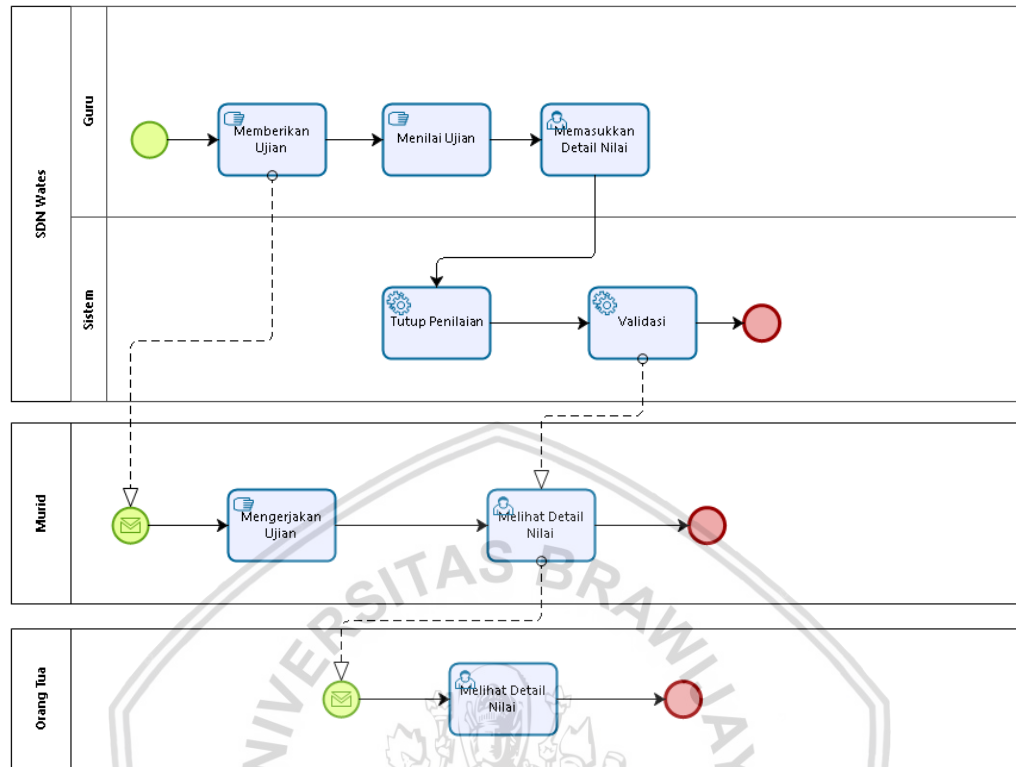
**Gambar 4.3 Proses Bisnis Mengevaluasi Murid Saat Ini**

Uraian proses bisnis mengevaluasi murid saat ini yang terdapat pada gambar 4.3 ialah sebagai berikut:

1. Proses bisnis dimulai dengan aktivitas guru memberikan ujian.
2. Murid mengerjakan ujian yang diberikan guru.
3. Guru kemudian menilai ujian dan mengembalikan lembar ujian ke murid.
4. Murid menerima nilai.
5. Guru merekap nilai kemudian menyerahkan laporan rekapitulasi ke kepala sekolah.
6. Kepala sekolah menerima laporan rekapitulasi nilai murid.



Sedangkan proses Mengevaluasi Murid usulan, dapat dilihat pada gambar 4.4



**Gambar 4.4 Proses Bisnis Mengevaluasi Murid Usulan**

Uraian proses bisnis mengevaluasi murid usulan yang terdapat pada gambar 4.4 ialah sebagai berikut:

1. Proses bisnis dimulai dengan aktivitas guru memberikan ujian.
2. Murid mengerjakan ujian yang diberikan guru.
3. Guru menilai ujian kemudian memasukkan detail nilai ke sistem pada tanggal yang ditentukan.
4. Sistem memberlakukan tutup penilaian kemudian sistem melakukan validasi nilai yang diisi.
5. Murid dan orang tua dapat melihat detail nilai pada sistem.

Aktivitas proses bisnis usulan dikodekan dengan BP-SIAS-x, dan diidentifikasi aktivitas proses bisnis mana yang akan difasilitasi sistem informasi yang akan dibuat. Tabel 4.2 menunjukkan kodifikasi dari aktivitas proses bisnis yang akan difasilitasi oleh sistem informasi.

**Tabel 4.2 Kodifikasi Aktivitas Proses Bisnis**

Proses Bisnis	Aktivitas	Kode Aktivitas
Registrasi	Mengisi Formulir Pendaftaran Murid	BP-SIAS-1
Registrasi	Melakukan Verifikasi Pendaftaran	BP-SIAS-2
Mengevaluasi Murid	Memasukkan Detail Nilai	BP-SIAS-3
Mengevaluasi Murid	Melihat Detail Nilai	BP-SIAS-4

### 4.3 Identifikasi Fitur

Fitur merepresentasikan solusi yang ditawarkan, dan dijelaskan pada tabel 4.3.

**Tabel 4.3 Identifikasi Fitur**

Kode Fitur	Fitur	Deskripsi
FEAT-SIAS-1	Mendaftar Murid	Sistem dapat digunakan untuk mendaftar murid baru
FEAT-SIAS-2	Memverifikasi Pendaftaran	Sistem dapat digunakan untuk memverifikasi pendaftaran murid baru
FEAT-SIAS-3	Memasukkan Detail Nilai	Sistem dapat digunakan untuk memasukkan detail nilai murid
FEAT-SIAS-4	Melihat Detail Nilai	Sistem dapat digunakan untuk melihat detail nilai murid
FEAT-SIAS-5	Autentikasi	Sistem dapat digunakan untuk mengautentikasi pengguna

#### 4.4 Identifikasi Aktor

Aktor-aktor yang terlibat dalam sistem diidentifikasi berdasarkan hasil pemodelan proses bisnis, dan dijelaskan dalam tabel 4.4.

**Tabel 4.4 Identifikasi Aktor**

Kode Aktor	Pool / Lane	Aktor	Deskripsi
A-SIAS-1	Petugas TU	Petugas TU	Petugas TU adalah pekerja / pegawai SDN Wates
A-SIAS-2	Guru	Guru	Guru adalah tenaga pengajar SDN Wates
A-SIAS-3	Murid	Murid	Murid adalah peserta didik SDN Wates
A-SIAS-4	Orang Tua	Orang Tua	Orang tua adalah orang tua dari murid SDN Wates

#### 4.5 Kebutuhan Fungsional

Kebutuhan Fungsional didapatkan berdasarkan hasil identifikasi fitur, dan dinyatakan dalam *software requirement statement* seperti yang tertera pada tabel 4.5.

**Tabel 4.5 Kebutuhan Fungsional**

Kode SRS	Kode Fitur	Kode Aktor	<i>Software Requirement Statement</i>	<i>Level</i>
F-SIAS-1	FEAT-SIAS-1	A-SIAS-4	Sistem harus bisa menampilkan formulir untuk menangani pendaftaran murid	Normal
F-SIAS-2	FEAT-SIAS-2	A-SIAS-2	Sistem harus bisa menangani verifikasi pendaftaran murid	Normal
F-SIAS-5	FEAT-SIAS-3	A-SIAS-2	Sistem harus bisa menampilkan formulir untuk menangani pemasukan detail nilai murid	Normal
F-SIAS-6	FEAT-SIAS-4	A-SIAS-3 A-SIAS-4	Sistem harus bisa menampilkan daftar detail nilai murid	Expected
F-SIAS-9	FEAT-SIAS-5	A-SIAS-1 A-SIAS-2 A-SIAS-3 A-SIAS-4	Sistem harus bisa menampilkan form untuk memasukkan nomor induk dan password, guna masuk kedalam sistem	Expected

#### 4.6 Kebutuhan Non-Fungsional

Kebutuhan Non-Fungsional dinyatakan dalam *software requirement statement* seperti yang tertera pada tabel 4.6 berikut dibawah ini:

**Tabel 4.6 Kebutuhan Non-Fungsional**

Kode SRS	<i>Software Requirement Statement</i>	<i>Level</i>
NF-SIAS-1	<i>Reliability</i> - Sistem dapat berfungsi dengan baik	Normal

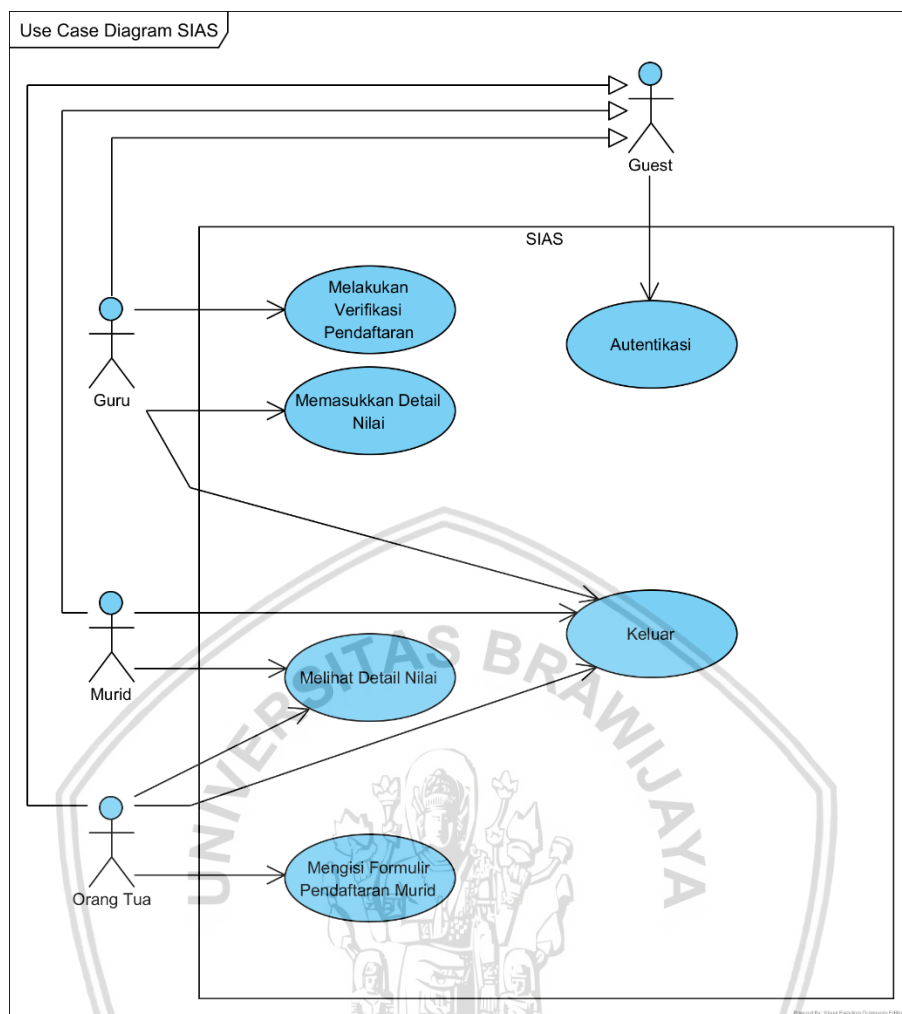
#### 4.7 Use Case Diagram

Berdasarkan kebutuhan fungsional yang telah didefinisikan, dapat diidentifikasi *Use Case* yang dituliskan pada tabel 4.7

**Tabel 4.7 Identifikasi Use Case**

Kode Use Case	Kode SRS	Use Case
UC-SIAS-1	F-SIAS-1	Mengisi Formulir Pendaftaran Murid
UC-SIAS-2	F-SIAS-2	Melakukan Verifikasi Pendaftaran
UC-SIAS-5	F-SIAS-5	Memasukkan Detail Nilai
UC-SIAS-6	F-SIAS-6	Melihat Detail Nilai
UC-SIAS-9	F-SIAS-9	Autentikasi
UC-SIAS-10	-	Keluar

Dalam ilustrasi dari sudut pandang perlakuan aktor terhadap sistem, dapat digambarkan menggunakan *use case diagram* pada gambar 4.5



Gambar 4.5 Use Case Diagram

## 4.8 Use Case Scenario

Use case yang telah dianalisis, akan dijelaskan alur jalannya dari sisi aktor dan sistem, menggunakan *use case senario*.

### 4.8.1 Use Case Scenario Mengisi Formulir Pendaftaran Murid (UC-SIAS-1)

Tabel 4.8 Use Case Scenario Mengisi Formulir Pendaftaran Murid

Use Case Name	Mengisi Formulir Pendaftaran Murid
Brief Description	Merupakan proses untuk mendaftarkan calon murid, yang dilakukan oleh orang tua murid dengan cara mengisi formulir pendaftaran murid
Actor	Orang tua
Pre-Condition	Terhubung dengan layanan internet

Tabel 4.8 *Use Case Scenario* Mengisi Formulir Pendaftaran Murid (Lanjutan)

<i>Basic Flow</i>	<ol style="list-style-type: none"> <li>1. Orang tua mengakses sistem</li> <li>2. Sistem menampilkan halaman utama</li> <li>3. Orang tua mengakses menu registrasi</li> <li>4. Sistem menampilkan halaman formulir pendaftaran murid baru</li> <li>5. Orang tua mengisi seluruh isian formulir dan menyertakan berkas yang perlu diunggah</li> <li>6. Orang tua mengakses menu daftar</li> <li>7. Sistem menyimpan data registrasi</li> </ol>
<i>Alternative Flows</i>	<p><b>Alt1</b> Orang tua tidak mengisi formulir dengan lengkap</p> <p>Jika pada langkah 5 orangtua tidak mengisi formulir dengan lengkap, maka ketika proses submit, sistem akan menampilkan pesan untuk mengisikan bagian dari formulir yang kosong</p>
<i>Post Condition</i>	Penambahan pada data registrasi

#### 4.8.2 *Use Case Scenario* Melakukan Verifikasi Pendaftaran (UC-SIAS-2)

Tabel 4.9 *Use Case Scenario* Melakukan Verifikasi Pendaftaran

<i>Use Case Name</i>	Melakukan Verifikasi Pendaftaran
<i>Brief Description</i>	Merupakan proses untuk memeriksa formulir yang diajukan oleh orangtua untuk dipilih apakah data yang diisi telah memenuhi kriteria atau tidak.
<i>Actor</i>	Guru
<i>Pre-Condition</i>	Guru telah masuk kedalam sistem (autentikasi)
<i>Basic Flow</i>	<ol style="list-style-type: none"> <li>1. Guru mengakses menu registrasi</li> <li>2. Sistem menampilkan daftar data registrasi</li> <li>3. Guru melihat detail masing-masing data registrasi dengan mengakses menu tampilkan</li> <li>4. Sistem menampilkan detail data registrasi yang dipilih</li> <li>5. Guru mengakses menu verifikasi</li> <li>6. Sistem mengubah status data registrasi terpilih menjadi sudah diverifikasi.</li> <li>7. Sistem menampilkan daftar data registrasi</li> </ol>



**Tabel 4.9 Use Case Scenario Melakukan Verifikasi Pendaftaran (Lanjutan)**

<i>Alternative Flows</i>	<b>Alt1</b> Guru tidak mengakses menu verifikasi Jika pada langkah 5 guru tidak mengakses menu verifikasi, maka sistem tidak mengubah status registrasi
<i>Post Condition</i>	Penambahan pada data murid dan orang tua

**4.8.3 Use Case Scenario Memasukkan Detail Nilai (UC-SIAS-5)****Tabel 4.10 Use Case Scenario Memasukkan Detail Nilai**

<i>Use Case Name</i>	Memasukkan Detail Nilai
<i>Brief Description</i>	Merupakan proses untuk memasukkan detail nilai kedalam sistem, sehingga dapat disimpan dan diakses
<i>Actor</i>	Guru
<i>Pre-Condition</i>	Guru telah masuk kedalam sistem (autentikasi)
<i>Basic Flow</i>	<ol style="list-style-type: none"> <li>1. Guru mengakses menu murid</li> <li>2. Sistem menampilkan halaman murid</li> <li>3. Guru memilih murid yang dituju</li> <li>4. Guru mengakses menu isi nilai</li> <li>5. Sistem menampilkan data murid yang dipilih dengan rincian isian komponen nilai</li> <li>6. Guru mengisi seluruh komponen nilai dengan lengkap</li> <li>7. Guru mengakses menu simpan</li> <li>8. Sistem menyimpan nilai yang diisi</li> <li>9. Sistem menampilkan halaman murid</li> </ol>
<i>Alternative Flows</i>	-
<i>Post Condition</i>	Penambahan pada data nilai

**4.8.4 Use Case Scenario Melihat Detail Nilai (UC-SIAS-6)****Tabel 4.11 Use Case Scenario Melihat Detail Nilai**

<i>Use Case Name</i>	Melihat Detail Nilai
<i>Brief Description</i>	Merupakan proses untuk menampilkan detail nilai beserta komponennya
<i>Actor</i>	Murid / Orang Tua

**Tabel 4.11 Use Case Scenario Melihat Detail Nilai (Lanjutan)**

<i>Pre-Condition</i>	Murid / Orang Tua telah masuk kedalam sistem (autentikasi), dan nilai telah diisi oleh guru
<i>Basic Flow</i>	<ol style="list-style-type: none"> <li>1. Murid / Orang Tua mengakses menu nilai</li> <li>2. Sistem menampilkan halaman rincian nilai</li> <li>3. Murid / Orang Tua melihat rincian nilai</li> </ol>
<i>Alternative Flows</i>	-
<i>Post Condition</i>	-

**4.8.5 Use Case Scenario Autentikasi (UC-SIAS-9)****Tabel 4.12 Use Case Scenario Autentikasi**

<i>Use Case Name</i>	Autentikasi
<i>Brief Description</i>	Merupakan proses untuk masuk kedalam sistem
<i>Actor</i>	Guest
<i>Pre-Condition</i>	Guest telah terdaftar sebagai guru, murid, petugas TU, maupun orang tua
<i>Basic Flow</i>	<ol style="list-style-type: none"> <li>1. Guest mengakses sistem</li> <li>2. Sistem menampilkan halaman utama</li> <li>3. Guest mengakses menu login</li> <li>4. Sistem menampilkan form login</li> <li>5. Guest mengisi form login</li> <li>6. Guest mengakses menu masuk</li> </ol>
<i>Alternative Flows</i>	<p><b>Alt1</b> Guest mengisi form login dengan data guru</p> <p>Jika pada langkah 5 guest mengisi form login dengan nip dan password guru, maka ketika mengakses menu masuk, sistem akan menampilkan halaman dasbor guru</p> <p><b>Alt2</b> Guest mengisi form login dengan data petugas TU</p> <p>Jika pada langkah 5 guest mengisi form login dengan nip dan password petugas TU, maka ketika mengakses menu masuk, sistem akan menampilkan halaman dasbor petugas TU</p>

Tabel 4.12 *Use Case Scenario* Autentikasi (Lanjutan)

<i>Alternative Flows</i>	<p><b>Alt3</b> Guest mengisi form login dengan data murid</p> <p>Jika pada langkah 5 guest mengisi form login dengan nisn dan password murid, maka ketika mengakses menu masuk, sistem akan menampilkan halaman dasbor murid</p> <p><b>Alt4</b> Guest mengisi form login dengan data orang tua</p> <p>Jika pada langkah 5 guest mengisi form login dengan nomor KK dan password orang tua, maka ketika mengakses menu masuk, sistem akan menampilkan halaman dasbor orang tua</p> <p><b>Alt5</b> Guest mengisi form login dengan data yang salah</p> <p>Jika pada langkah 5 guest mengisi form login dengan nomor induk dan password yang salah, maka ketika mengakses menu masuk, sistem akan menampilkan halaman utama (home)</p>
<i>Post Condition</i>	-

#### 4.8.6 *Use Case Scenario* Keluar (UC-SIAS-10)

Tabel 4.13 *Use Case Scenario* Keluar

<i>Use Case Name</i>	Keluar
<i>Brief Description</i>	Merupakan proses untuk keluar dari sistem
<i>Actor</i>	Guru / Petugas TU / Murid / Orang tua
<i>Pre-Condition</i>	Guru / Petugas TU / Murid / Orang tua telah masuk kedalam sistem (autentikasi)
<i>Basic Flow</i>	<ol style="list-style-type: none"> <li>1. Guru / Petugas TU / Murid / Orang tua mengakses menu logout</li> <li>2. Sistem menampilkan halaman utama (home)</li> </ol>
<i>Alternative Flows</i>	-
<i>Post Condition</i>	-

## BAB 5 PERANCANGAN DAN IMPLEMENTASI

Perancangan merupakan visualisasi dari sistem, yang sebelumnya telah melalui tahap analisis kebutuhan. Rancangan sistem adalah *blueprint* untuk implementasi (kodifikasi). Orientasi perancangan yang dipakai adalah berbasis *object* sehingga menggunakan *UML (Unified Modeling Language)* untuk bahasa standar rancangan, dan divisualisasikan dengan *activity diagram*, *sequence diagram*, *class diagram*, *physical data model*, dan sketsa UI.

### 5.1 Analisis Inkremental

Analisis Inkremental dilakukan untuk menentukan inkremen berdasarkan prioritas kebutuhan yang telah didefinisikan pada tahap analisis kebutuhan. Hasil dari analisis inkremental digunakan sebagai acuan untuk urutan dari pengembangan. Pengembangan pada masing-masing fitur meliputi perancangan *Activity Diagram*, dan perancangan *Sequence Diagram*. Tabel 5.1 menyajikan prioritas serta fitur mana yang akan dirancang terlebih dahulu.

**Tabel 5.1 Tabel Analisis Inkremental**

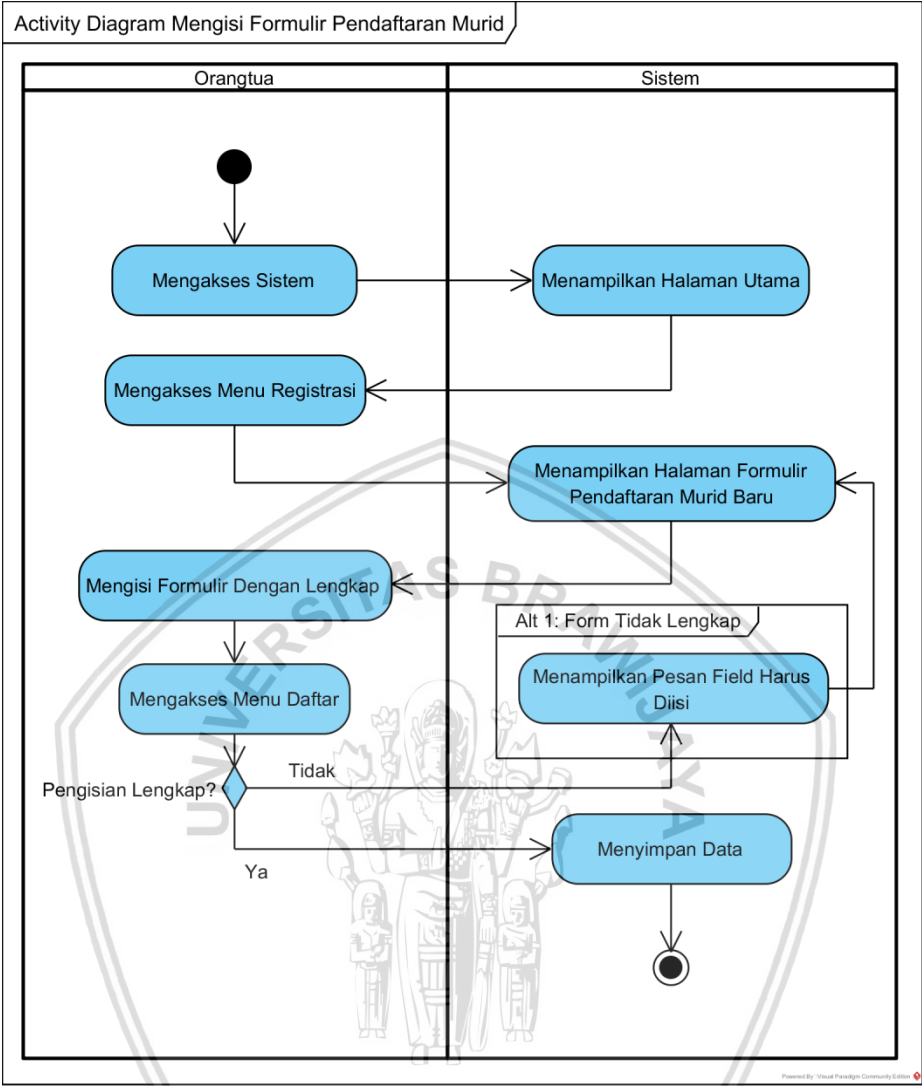
Kode SRS	Level	Kode Use Case	Nama Fitur	Urutan Inkremen
F-SIAS-1	Normal	UC-SIAS-1	Mengisi Formulir Pendaftaran Murid	Inkremen Pertama
F-SIAS-2	Normal	UC-SIAS-2	Melakukan Verifikasi Pendaftaran	Inkremen Kedua
F-SIAS-5	Normal	UC-SIAS-5	Memasukkan Detail Nilai	Inkremen Ketiga

### 5.2 Inkremen Pertama

Inkremen Pertama dilakukan dengan memodelkan skenario dari *use case* kedalam *Activity Diagram* untuk mengetahui bagaimana interaksi antara pengguna dengan sistem, serta memodelkan *Sequence Diagram* untuk mengetahui bagaimana pertukaran pesan yang dilakukan antar komponen dalam sistem.

Berdasarkan Analisis Inkremental, diagram aktivitas dan diagram sekuen yang dimodelkan yaitu pada fitur Mengisi Formulir Pendaftaran Murid.

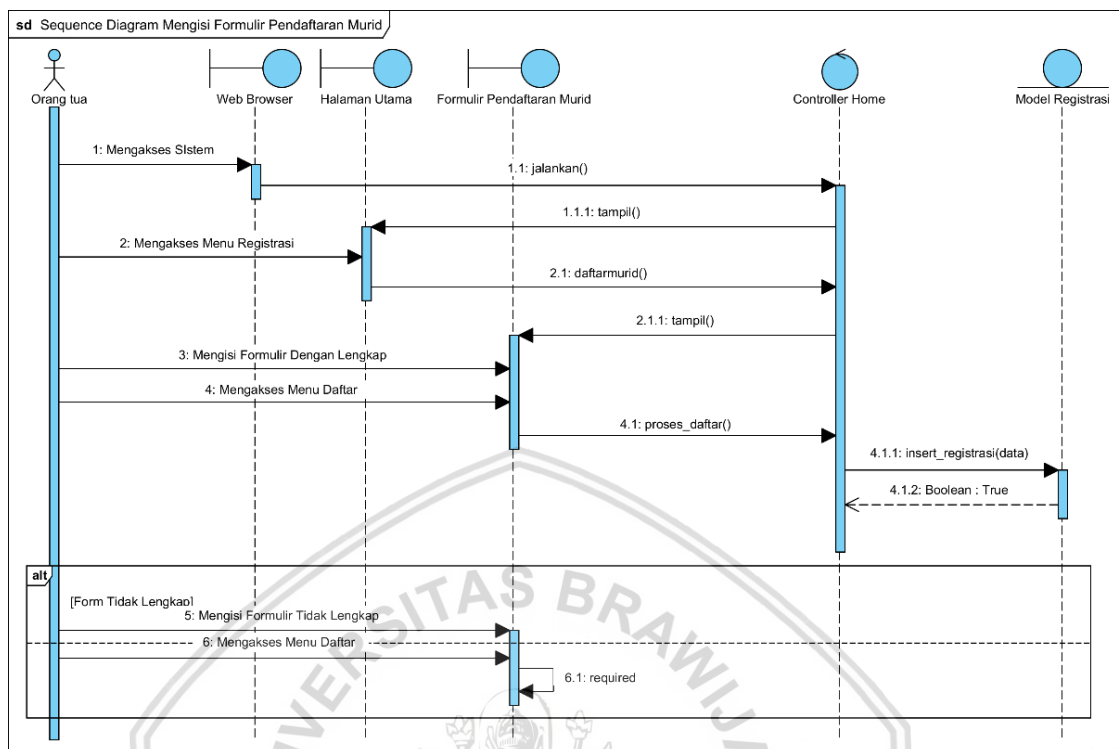
5.2.1 Activity Diagram Mengisi Formulir Pendaftaran Murid (AD-SIAS-1)



Gambar 5.1 Activity Diagram Mengisi Formulir Pendaftaran Murid

Gambar 5.1 menjelaskan alur aktivitas dimana orang tua melakukan pengisian formulir pendaftaran murid sesuai dengan *use case scenario* yang telah dibuat. Sehingga melalui proses pengisian formulir, didapatkan data registrasi untuk dimanfaatkan pada proses verifikasi.

## 5.2.2 Sequence Diagram Mengisi Formulir Pendaftaran Murid (SQ-SIAS-1)



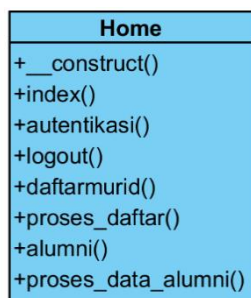
**Gambar 5.2 Sequence Diagram Mengisi Formulir Pendaftaran Murid**

Gambar 5.2 menjelaskan alur mekanisme interaksi sekuensial antar aktor orang tua, dengan *controller home* yang menangani permintaan aktor orang tua, dengan *boundary web browser* untuk mengakses sistem, *boundary halaman utama* untuk menampilkan menu, *boundary formulir pendaftaran murid* untuk mengisi data, serta *model registrasi* untuk menyimpan data.

## 5.2.3 Pemodelan Class Diagram

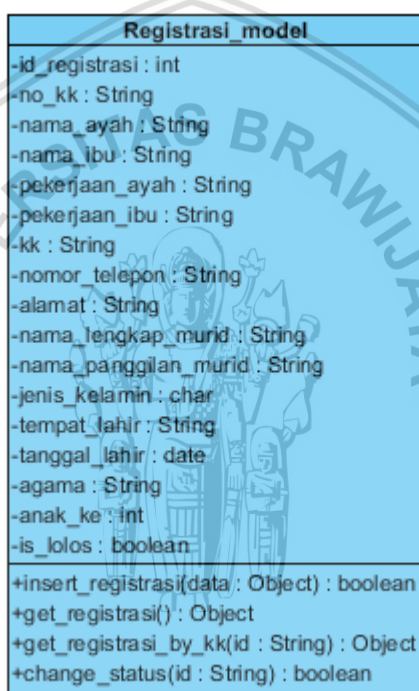
Diagram kelas dimodelkan berdasarkan diagram sekuen, yang terdapat pada fitur mengisi formulir pendaftaran murid. Dengan menggunakan prinsip MVC (*Model – View – Controller*), diagram kelas dapat digolongkan menjadi 2 jenis diagram kelas yakni diagram kelas sebagai *logical class* dan diagram kelas sebagai *domain model*.





**Gambar 5.3 Class Diagram Home sebagai Logical Class**

Kelas *Home* yang terdapat pada gambar 5.3, merupakan kelas utama yang mengelola tampilan pengguna untuk keperluan akses sistem, dan menampilkan formulir.

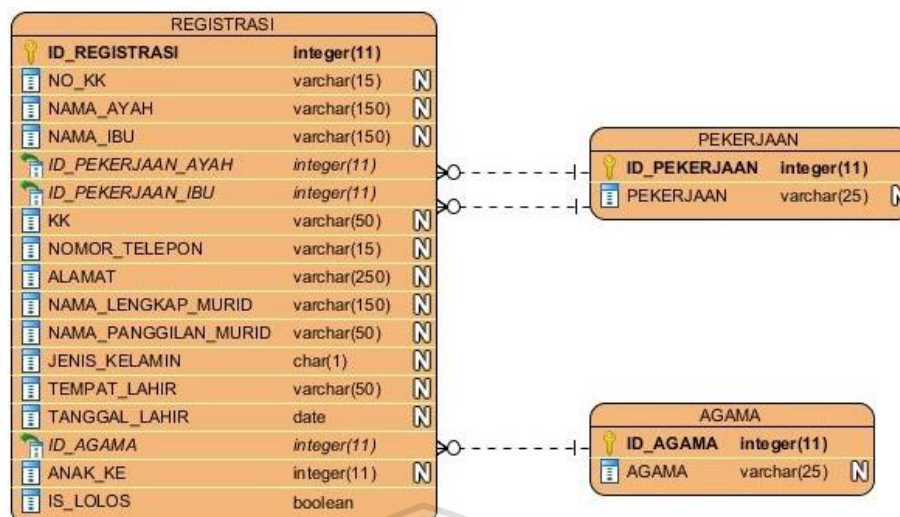


**Gambar 5.4 Class Diagram Registrasi\_model sebagai Domain Model**

Kelas *Registrasi\_model* yang terdapat pada gambar 5.4, merupakan kelas yang akan digunakan untuk acuan pemodelan data.

#### 5.2.4 Pemodelan Data

Data dimodelkan dalam bentuk *Physical Data Modeling*, yang dapat dilihat pada gambar 5.5



**Gambar 5.5 Physical Data Modeling**

Gambar 5.5 menggambarkan model tabel yang telah dilakukan normalisasi, kemudian dispesifikasikan menjadi tiga tabel di bawah ini.

#### 5.2.4.1 Tabel Agama

**Tabel 5.2 Tabel Agama**

No	Nama Kolom	Tipe Data	Ukuran	Constraint
1	ID_AGAMA	INTEGER	-	Primary Key
2	AGAMA	VARCHAR	25	-

#### 5.2.4.2 Tabel Pekerjaan

**Tabel 5.3 Tabel Pekerjaan**

No	Nama Kolom	Tipe Data	Ukuran	Constraint
1	ID_Pekerjaan	INTEGER	-	Primary Key
2	PEKERJAAN	VARCHAR	25	-

#### 5.2.4.3 Tabel Registrasi

**Tabel 5.4 Tabel Registrasi**

No	Nama Kolom	Tipe Data	Ukuran	Constraint
1	ID_REGISTRASI	INTEGER	-	Primary Key
2	NO_KK	VARCHAR	15	-
3	NAMA_AYAH	VARCHAR	150	-

**Tabel 5.4 Tabel Registrasi (lanjutan)**

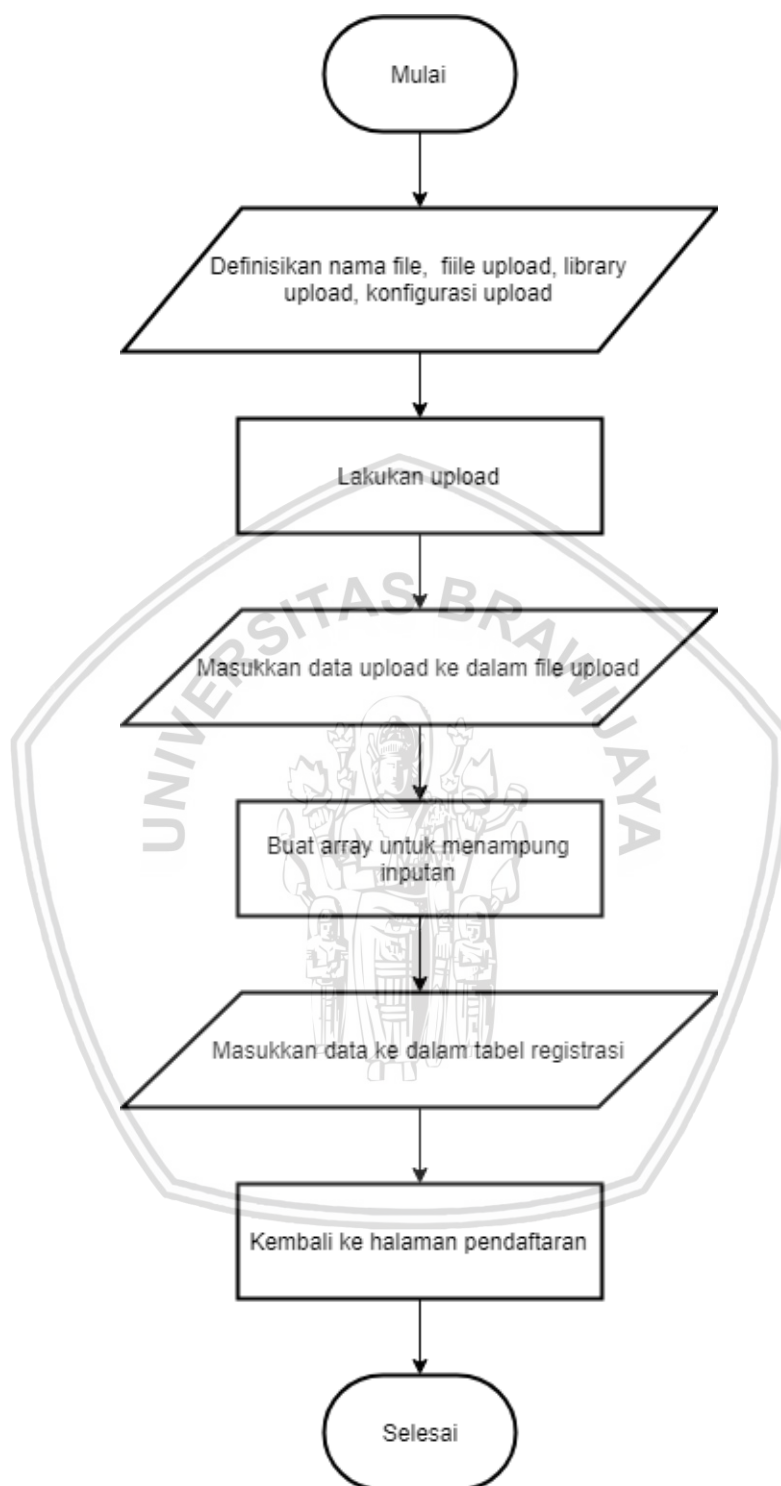
No	Nama Kolom	Type Data	Ukuran	Constraint
4	NAMA_IBU	VARCHAR	150	-
5	ID_PKERJAAN_AYAH	INTEGER	11	Foreign Key
6	ID_PKERJAAN_IBU	INTEGER	11	Foreign Key
7	KK	VARCHAR	50	-
8	NOMOR_TELEPON	VARCHAR	15	-
9	ALAMAT	VARCHAR	250	-
10	NAMA LENGKAP_MURID	VARCHAR	150	-
11	NAMA_PANGGILAN_MURID	VARCHAR	50	-
12	JENIS_KELAMIN	CHAR	1	-
13	TEMPAT_LAHIR	VARCHAR	50	-
14	TANGGAL_LAHIR	DATE	-	-
15	ID_AGAMA	INTEGER	-	Foreign Key
16	ANAK_KE	INTEGER	-	-
17	IS_LOLOS	BOOLEAN	-	Not Null

### 5.2.5 Desain Algoritma Pendaftaran Murid

Algoritma pendaftaran murid merupakan algoritma untuk memproses data yang dimasukkan pada formulir pendaftaran kedalam sistem. Tabel 5.5 dan gambar 5.6 menjelaskan bagaimana urutan dari proses pendaftaran murid.

**Tabel 5.5 Algoritma Pendaftaran Murid**

Baris	Proses_daftar()
1	Mulai
2	Definisikan nama file
3	Konfigurasi file upload
4	Instansiasi library upload
5	Inisialisasi konfigurasi upload
6	Lakukan upload
7	Masukkan data upload ke dalam file upload
8	Buat array untuk menampung semua inputan
9	Masukkan kedalam tabel registrasi
10	Kembali ke halaman pendaftaran murid
11	Selesai



**Gambar 5.6 Flowchart Pendaftaran Murid**

### 5.2.6 Desain Antarmuka Halaman Registrasi

**Formulir Pendaftaran**

Nomor KK

Nama Ayah

Nama Ibu

Pekerjaan Ayah

Pekerjaan Ibu

Nomor Telepon

Nama Lengkap Murid

Nama Panggilan

Jenis Kelamin

Tempat, Lahir

Tanggal Lahir

Agama

Anak Ke

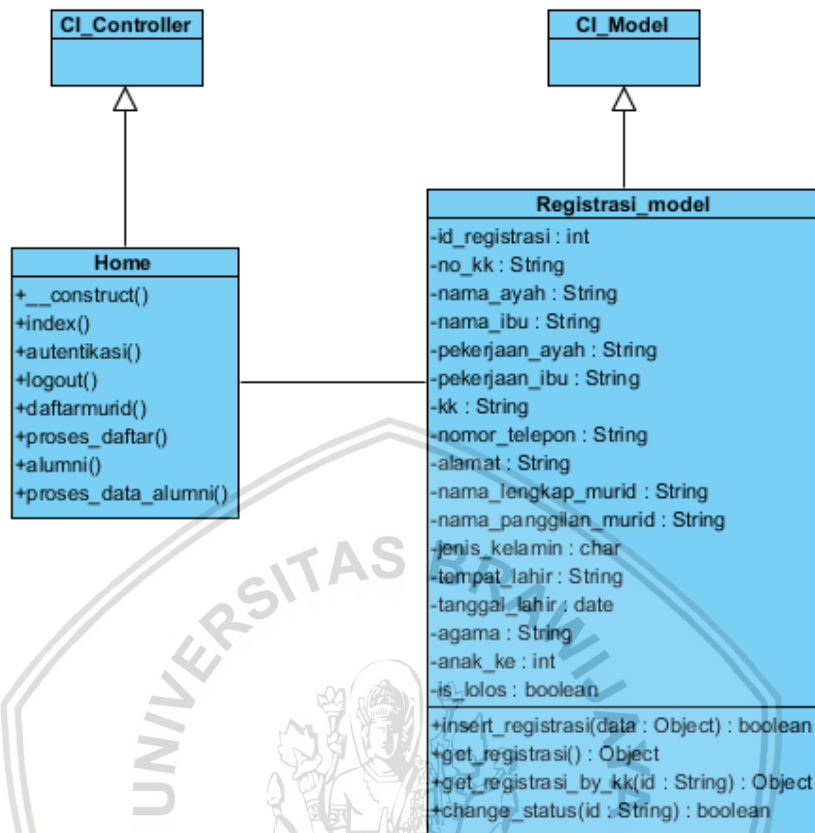
Alamat

Unggah Foto Kartu Keluarga

**Gambar 5.7 Desain Antarmuka Halaman Registrasi**

Desain Antarmuka Halaman registrasi yang terdapat pada gambar 5.7 merupakan desain antarmuka halaman untuk memasukkan data pendaftaran murid, dan terdapat tombol daftar untuk mengeksekusi pendaftaran murid.

### 5.2.7 Implementasi Class Diagram



Gambar 5.8 Implementasi Class Diagram

Gambar 5.8 merupakan diagram kelas yang telah diimplementasikan menggunakan *framework CodeIgniter*. Terdapat satu kelas *controller* yang diimplementasikan, yaitu *controller Home*, serta terdapat satu kelas model yaitu *Registrasi\_model*.

### 5.2.8 Implementasi DDL (Data Definition Language)

Berikut merupakan implementasi DDL (*Data Definition Language*) berdasarkan rancangan, untuk penyimpanan data pada sistem, yang dapat dilihat pada tabel 5.6.

Tabel 5.6 Implementasi DDL

```

/* 1. Tabel Agama */
CREATE TABLE AGAMA (
    ID_AGAMA int(11) NOT NULL PRIMARY KEY,
    AGAMA varchar(25)
);
    
```



**Tabel 5.6 Implementasi DDL (Lanjutan)**

```

/* 2. Tabel Pekerjaan */
CREATE TABLE PEKERJAAN (
    ID_PKERJAAN int(11) NOT NULL PRIMARY KEY,
    PEKERJAAN varchar(25)
);

/* 3. Tabel Registrasi */
CREATE TABLE REGISTRASI (
    ID_REGISTRASI int(11) NOT NULL PRIMARY KEY,
    NO_KK varchar(15),
    NAMA_AYAH varchar(150),
    NAMA_IBU varchar(150),
    ID_PKERJAAN_AYAH int(11),
    FOREIGN KEY (ID_PKERJAAN_AYAH) REFERENCES PEKERJAAN
    (ID_PKERJAAN),
    PEKERJAAN_IBU int(11),
    FOREIGN KEY (ID_PKERJAAN_IBU) REFERENCES PEKERJAAN
    (ID_PKERJAAN),
    KK varchar(50),
    NOMOR_TELEPON varchar(15),
    ALAMAT varchar(250),
    NAMA LENGKAP_MURID varchar(150),
    NAMA_PANGGILAN_MURID varchar(50),
    JENIS_KELAMIN char(1),
    TEMPAT_LAHIR varchar(50),
    TANGGAL_LAHIR date,
    ID_AGAMA int(11) NOT NULL,
    FOREIGN KEY (ID_AGAMA) REFERENCES AGAMA (ID_AGAMA),
    ANAK_KE int(11),
    IS_LOLOS boolean NOT NULL DEFAULT 'False'
);

```

### 5.2.9 Implementasi Fungsi Pendaftaran Murid

Fungsi pendaftaran murid terdapat pada kelas *Controller Home*, yangmana merupakan fungsi untuk memasukkan data dari formulir pendaftaran, kedalam data registrasi. Tabel 5.7 menunjukkan bagaimana kode sumber dari fungsi pendaftaran murid.

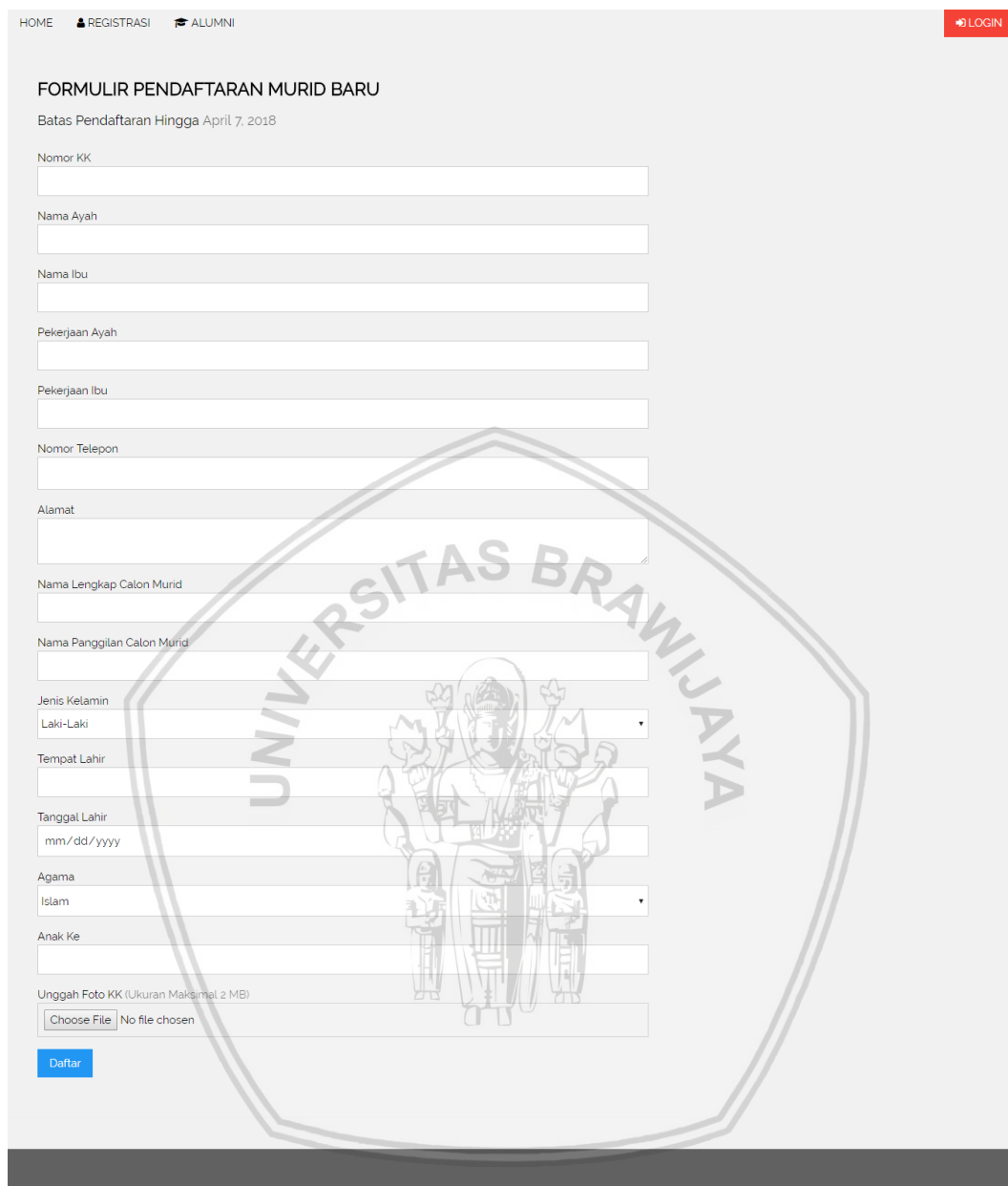
**Tabel 5.7 Implementasi Fungsi Pendaftaran Murid**

```

public function proses_daftar()
{
    $nama_file = "Foto_KK".time();
    $config['upload_path'] = './assets/images/';
    $config['allowed_types'] = 'gif|jpg|png|jpeg|bmp';
    $config['max_size'] = 0;
    $config['max_width'] = 0;
    $config['max_height'] = 0;
    $config['file_name'] = $nama_file;
    $this->load->library('upload', $config);
    $this->upload->initialize($config);
    $this->upload->do_upload('Upload_Foto_KK');
    $file_upload = $this->upload->data();
    $data = array(
        'no_kk' => $this->input->post('No_KK'),
        'nama_ayah' => $this->input->post('Nama_Ayah'),
        'nama_ibu' => $this->input->post('Nama_Ibu'),
        'pekerjaan_ayah' => $this->input->post('Pekerjaan_Ayah'),
        'pekerjaan_ibu' => $this->input->post('Pekerjaan_Ibu'),
        'kk' => $file_upload['file_name'],
        'nomor_telepon' => $this->input->post('Nomor_Telepon'),
        'alamat' => $this->input->post('Alamat'),
        'nama_lengkap_murid' => $this->input->post('Nama_Lengkap'),
        'nama_panggilan_murid' => $this->input->post('Nama_Panggilan'),
        'jenis_kelamin' => $this->input->post('Jenis_Kelamin'),
        'tempat_lahir' => $this->input->post('Tempat_Lahir'),
        'tanggal_lahir' => $this->input->post('Tanggal_Lahir'),
        'agama' => $this->input->post('Agama'),
        'anak_ke' => $this->input->post('Anak_Ke')
    );
    $this->Registrasi_model->insert_registrasi($data);
    redirect(base_url('Home/daftarmurid'));
}

```

### 5.2.10 Implementasi Antarmuka Halaman Registrasi



HOME REGISTRASI ALUMNI LOGIN

**FORMULIR PENDAFTARAN MURID BARU**

Batas Pendaftaran Hingga April 7, 2018

Nomor KK

Nama Ayah

Nama Ibu

Pekerjaan Ayah

Pekerjaan Ibu

Nomor Telepon

Alamat

Nama Lengkap Calon Murid

Nama Panggilan Calon Murid

Jenis Kelamin

Tempat Lahir

Tanggal Lahir

Agama

Anak Ke

Unggah Foto KK (Ukuran Maksimal 2 MB)  
 No file chosen

**Gambar 5.9 Antarmuka Halaman Registrasi**

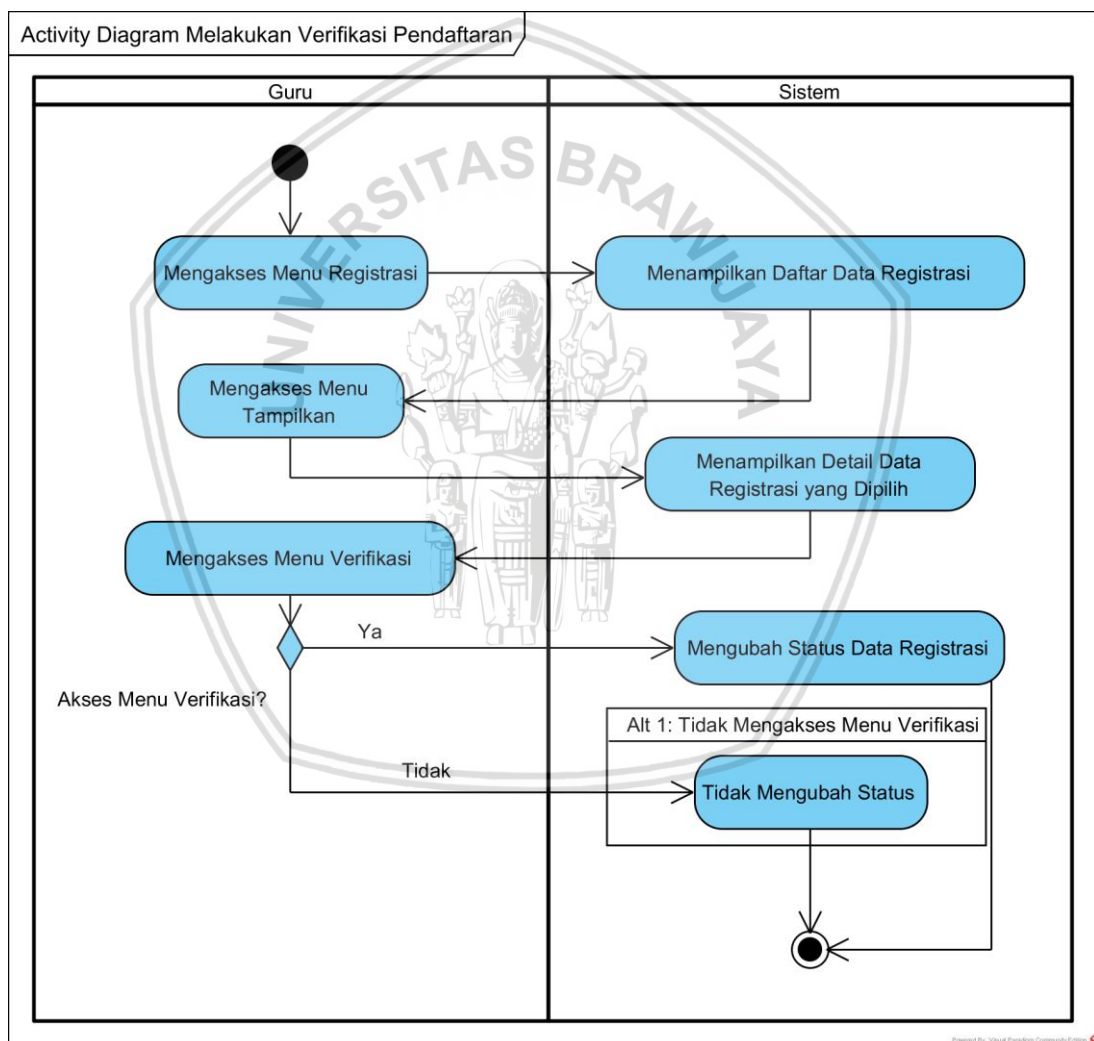
Halaman registrasi yang terdapat pada gambar 5.9 merupakan halaman untuk memasukkan data pendaftaran murid, dan terdapat tombol daftar untuk mengeksekusi pendaftaran murid.

### 5.3 Inkremen Kedua

Inkremen Kedua dilakukan dengan memodelkan skenario dari *use case* kedalam *Activity Diagram* untuk mengetahui bagaimana interaksi antara pengguna dengan sistem, serta memodelkan *Sequence Diagram* untuk mengetahui bagaimana pertukaran pesan yang dilakukan antar komponen dalam sistem.

Berdasarkan Analisis Inkremental, diagram aktivitas dan diagram sekuen yang dimodelkan yaitu pada fitur Melakukan Verifikasi Pendaftaran.

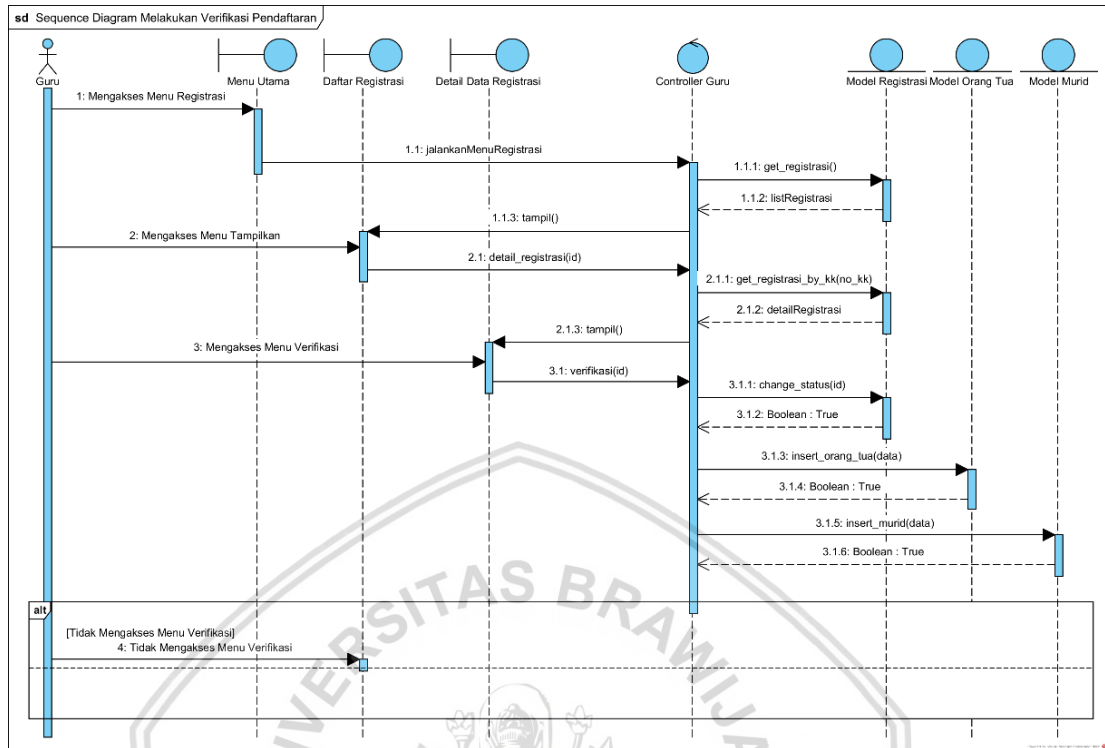
#### 5.3.1 Activity Diagram Melakukan Verifikasi Pendaftaran (AD-SIAS-2)



**Gambar 5.10 Activity Diagram Melakukan Verifikasi Pendaftaran**

Gambar 5.10 menjelaskan alur aktivitas dimana guru melakukan verifikasi pendaftaran terhadap data registrasi sesuai dengan *use case scenario* yang telah dibuat. Sehingga melalui proses verifikasi didapatkan data orangtua, serta data murid yang lolos seleksi pendaftaran.

### 5.3.2 Sequence Diagram Melakukan Verifikasi Pendaftaran (SQ-SIAS-2)



**Gambar 5.11 Sequence Diagram Melakukan Verifikasi Pendaftaran**

Gambar 5.11 menjelaskan alur mekanisme interaksi sekuensial antar aktor guru, dengan *controller guru* yang menangani permintaan aktor guru, dengan *boundary* menu utama untuk mengakses menu yang lain, *boundary* daftar registrasi untuk menampilkan data dari seluruh registrasi, *boundary* detail data registrasi untuk menampilkan detail dari data registrasi yang dipilih, serta *model* registrasi, *model* orang tua, dan *model* murid untuk mengelola data.

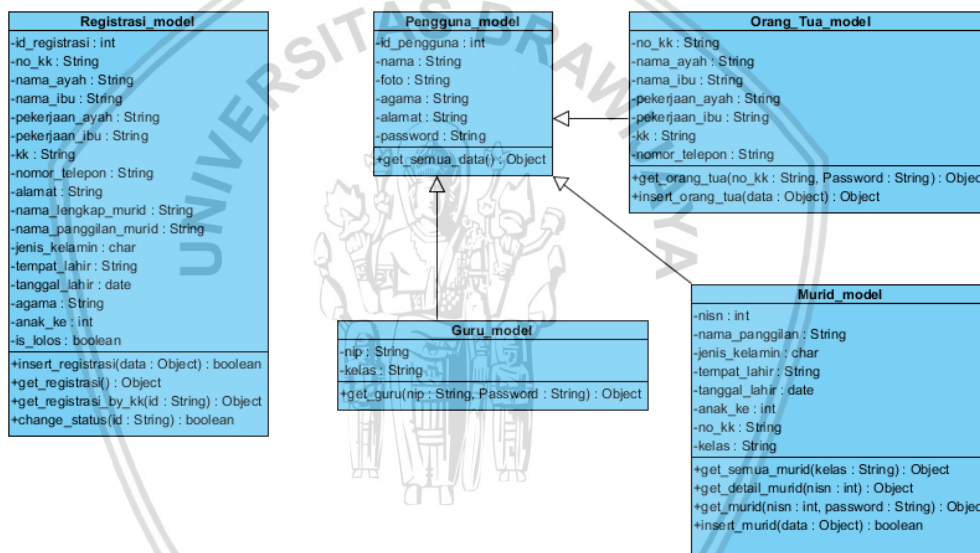
### 5.3.3 Pemodelan Class Diagram

Diagram kelas dimodelkan berdasarkan diagram sekuen, yang terdapat pada fitur mengisi formulir pendaftaran murid. Dengan menggunakan prinsip MVC (*Model – View – Controller*), diagram kelas dapat digolongkan menjadi 2 jenis diagram kelas yakni diagram kelas sebagai *logical class* dan diagram kelas sebagai *domain model*.



**Gambar 5.12 Class Diagram Guru sebagai Logical Class**

Kelas Guru yang terdapat pada gambar 5.12, merupakan kelas untuk mengelola data murid, dan registrasi, serta menampilkan formulir.



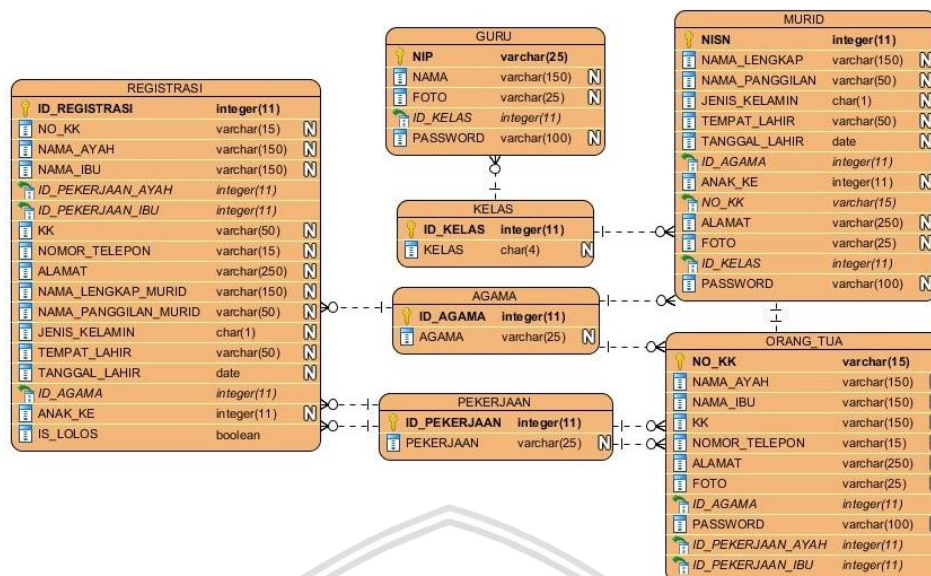
**Gambar 5.13 Class Diagram sebagai Domain Model**

Kelas Registrasi\_model, Pengguna\_model, Guru\_model, Murid\_model, dan Orang\_Tua\_model yang terdapat pada gambar 5.13, merupakan kelas untuk acuan pemodelan data.

### 5.3.4 Pemodelan Data

Data dimodelkan dalam *Physical Data Modeling*, yang dapat dilihat pada gambar 5.14





**Gambar 5.14 Physical Data Modeling**

Gambar 5.14 menggambarkan model tabel, yang kemudian dispesifikasikan menjadi 7 tabel di bawah ini.

#### 5.3.4.1 Tabel Agama

**Tabel 5.8 Tabel Agama**

No	Nama Kolom	Tipe Data	Ukuran	Constraint
1	ID_AGAMA	INTEGER	-	Primary Key
2	AGAMA	VARCHAR	25	-

#### 5.3.4.2 Tabel Pekerjaan

**Tabel 5.9 Tabel Pekerjaan**

No	Nama Kolom	Tipe Data	Ukuran	Constraint
1	ID_PKERJAAN	INTEGER	-	Primary Key
2	PEKERJAAN	VARCHAR	25	-

#### 5.3.4.3 Tabel Kelas

**Tabel 5.10 Tabel Kelas**

No	Nama Kolom	Tipe Data	Ukuran	Constraint
1	ID_KELAS	INTEGER	-	Primary Key
2	KELAS	CHAR	4	-

## 5.3.4.4 Tabel Registrasi

Tabel 5.11 Tabel Registrasi

No	Nama Kolom	Tipe Data	Ukuran	Constraint
1	ID_REGISTRASI	INTEGER	-	Primary Key
2	NO_KK	VARCHAR	15	-
3	NAMA_AYAH	VARCHAR	150	-
4	NAMA_IBU	VARCHAR	150	-
5	ID_PKERJAAN_AYAH	INTEGER	-	Foreign Key
6	ID_PKERJAAN_IBU	INTEGER	-	Foreign Key
7	KK	VARCHAR	50	-
8	NOMOR_TELEPON	VARCHAR	15	-
9	ALAMAT	VARCHAR	250	-
10	NAMA LENGKAP_MURID	VARCHAR	150	-
11	NAMA_PANGGILAN_MURID	VARCHAR	50	-
12	JENIS_KELAMIN	CHAR	1	-
13	TEMPAT_LAHIR	VARCHAR	50	-
14	TANGGAL_LAHIR	DATE	-	-
15	ID_AGAMA	INTEGER	-	Foreign Key
16	ANAK_KE	INTEGER	-	-
17	IS_LOLOS	BOOLEAN	-	Not Null

## 5.3.4.5 Tabel Orang Tua

Tabel 5.12 Tabel Orang Tua

No	Nama Kolom	Tipe Data	Ukuran	Constraint
1	NO_KK	VARCHAR	15	Primary Key
2	NAMA_AYAH	VARCHAR	150	-
3	NAMA_IBU	VARCHAR	150	-
4	ID_PKERJAAN_AYAH	INTEGER	-	Foreign Key
5	ID_PKERJAAN_IBU	INTEGER	-	Foreign Key
6	KK	VARCHAR	50	-
7	NOMOR_TELEPON	VARCHAR	15	-

**Tabel 5.12 Tabel Orang Tua (lanjutan)**

No	Nama Kolom	Tipe Data	Ukuran	Constraint
8	ALAMAT	VARCHAR	250	-
9	FOTO	VARCHAR	25	Not Null
10	PASSWORD	VARCHAR	100	Not Null

**5.3.4.6 Tabel Murid****Tabel 5.13 Tabel Murid**

No	Nama Kolom	Tipe Data	Ukuran	Constraint
1	NISN	INTEGER	-	Primary Key
2	NAMA LENGKAP	VARCHAR	150	-
3	NAMA PANGGILAN	VARCHAR	50	-
4	JENIS KELAMIN	CHAR	1	-
5	TEMPAT LAHIR	VARCHAR	50	-
6	TANGGAL LAHIR	DATE	-	-
7	ID_AGAMA	INTEGER	-	Foreign Key
8	ANAK_KE	INTEGER	-	-
9	NO_KK	VARCHAR	15	Foreign Key
10	ALAMAT	VARCHAR	250	-
11	FOTO	VARCHAR	25	Not Null
12	ID_KELAS	INTEGER	-	Foreign Key
13	PASSWORD	VARCHAR	100	Not Null

**5.3.4.7 Tabel Guru****Tabel 5.14 Tabel Guru**

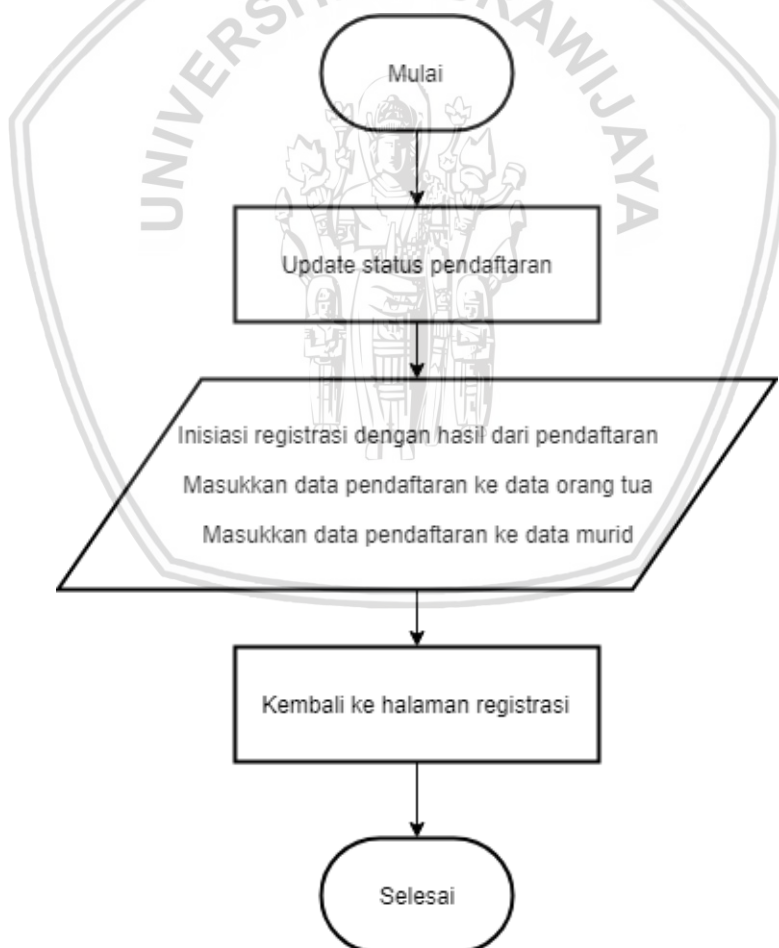
No	Nama Kolom	Tipe Data	Ukuran	Constraint
1	NIP	VARCHAR	25	Primary Key
2	NAMA	VARCHAR	150	-
3	FOTO	VARCHAR	25	Not Null
4	ID_KELAS	INTEGER	-	Foreign Key
5	PASSWORD	VARCHAR	100	Not Null

### 5.3.5 Verifikasi Pendaftaran

Algoritma pendaftaran murid merupakan algoritma untuk memverifikasi data murid, berdasarkan data registrasi. Tabel 5.15 dan gambar 5.15 menjelaskan bagaimana urutan dari proses verifikasi pendaftaran.

**Tabel 5.15 Algoritma Verifikasi Pendaftaran**

Baris	Verifikasi (id)
1	Mulai
2	Update status pendaftaran
3	Inisiasi registrasi dengan hasil dari pendaftaran
4	Masukkan data pendaftaran ke data orang tua
5	Masukkan data pendaftaran ke data murid
6	Kembali ke halaman registrasi
7	Selesai



**Gambar 5.15 Flowchart Verifikasi Pendaftaran**

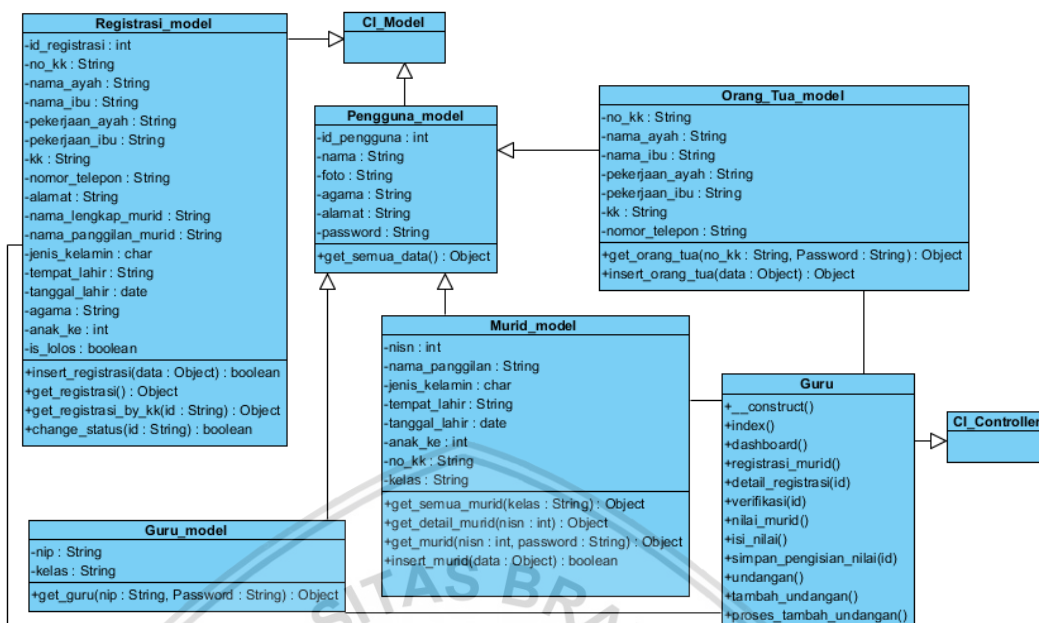
### 5.3.6 Desain Antarmuka Halaman Verifikasi Pendaftaran

Logo	<div> <input type="text"/> </div> <div> ID Registrasi  Nomor KK  Nama Ayah  Nama Ibu  Pekerjaan Ayah  Pekerjaan Ibu  Nomor Telepon  Nama Lengkap Murid  Nama Panggilan  Jenis Kelamin  Tempat, dan Tanggal Lahir  Agama  Anak Ke  Alamat  Foto Kartu Keluarga </div> <div> Tombol Verifikasi </div>
------	---

**Gambar 5.16 Desain Antarmuka Halaman Verifikasi Pendaftaran**

Desain Antarmuka Halaman Verifikasi pendaftaran merupakan desain antarmuka halaman untuk melakukan verifikasi pendaftaran murid baru. Terdapat tombol verifikasi untuk menampilkan memverifikasi data.

### 5.3.7 Implementasi Class Diagram



Gambar 5.17 Implementasi Class Diagram

Gambar 5.15 merupakan diagram kelas yang telah diimplementasikan menggunakan *framework CodeIgniter*. Terdapat satu kelas *controller* yang diimplementasikan, yaitu *controller* Guru, serta terdapat 5 kelas model yaitu Registrasi\_model, Pengguna\_model, Guru\_model, Murid\_model, dan Orang\_Tua\_model.

### 5.3.8 Implementasi DDL (Data Definition Language)

Berikut merupakan implementasi DDL (*Data Definition Language*) berdasarkan rancangan, untuk penyimpanan data pada sistem, yang dapat dilihat pada tabel 5.16.

Tabel 5.16 Implementasi DDL

```

/* 1. Tabel Agama */
CREATE TABLE AGAMA (
    ID_AGAMA int(11) NOT NULL PRIMARY KEY,
    AGAMA varchar(25)
);
/* 2. Tabel Pekerjaan */
CREATE TABLE PEKERJAAN (
    ID_Pekerjaan int(11) NOT NULL PRIMARY KEY,
    PEKERJAAN varchar(25)
);
  
```



**Tabel 5.16 Implementasi DDL (Lanjutan)**

```

/* 3. Tabel Kelas */
CREATE TABLE KELAS (
    ID_KELAS int(11) NOT NULL PRIMARY KEY,
    KELAS char(4)
);

/* 4. Tabel Registrasi */
CREATE TABLE REGISTRASI (
    ID_REGISTRASI int(11) NOT NULL PRIMARY KEY,
    NO_KK varchar(15),
    NAMA_AYAH varchar(150),
    NAMA_IBU varchar(150),
    ID_PKERJAAN_AYAH int(11),
    FOREIGN KEY (ID_PKERJAAN_AYAH) REFERENCES
    PEKERJAAN (ID_PKERJAAN),
    ID_PKERJAAN_IBU int(11),
    FOREIGN KEY (ID_PKERJAAN_IBU) REFERENCES
    PEKERJAAN (ID_PKERJAAN),
    KK varchar(50),
    NOMOR_TELEPON varchar(15),
    ALAMAT varchar(250),
    NAMA LENGKAP_MURID varchar(150),
    NAMA_PANGGILAN_MURID varchar(50),
    JENIS_KELAMIN char(1),
    TEMPAT_LAHIR varchar(50),
    TANGGAL_LAHIR date,
    ID_AGAMA int(11) NOT NULL,
    FOREIGN KEY (ID_AGAMA) REFERENCES AGAMA (ID_AGAMA),
    ANAK_KE int(11),
    IS_LOLOS boolean NOT NULL DEFAULT 'False'
);

/* 5. Tabel Orang Tua */
CREATE TABLE ORANG_TUA (
    NO_KK varchar(15) NOT NULL PRIMARY KEY,
    NAMA_AYAH varchar(150),
    NAMA_IBU varchar(150),
    ID_PKERJAAN_AYAH int(11),
    FOREIGN KEY (ID_PKERJAAN_AYAH) REFERENCES
    PEKERJAAN (ID_PKERJAAN),

```

**Tabel 5.16 Implementasi DDL (Lanjutan)**

```

ID_PKERJAAN_IBU int(11),
FOREIGN KEY (ID_PKERJAAN_IBU) REFERENCES
PEKERJAAN (ID_PKERJAAN),
KK varchar(50),
NOMOR_TELEPON varchar(15),
ALAMAT varchar(250),
FOTO varchar(25) NOT NULL DEFAULT 'Pengguna.jpg',
PASSWORD varchar(100) NOT NULL DEFAULT '123456'
);
/* 6. Tabel Murid */
CREATE TABLE MURID (
NISN int(11) NOT NULL PRIMARY KEY,
NAMA LENGKAP varchar(150),
NAMA PANGGILAN varchar(50),
JENIS_KELAMIN char(1),
TEMPAT_LAHIR varchar(50),
TANGGAL_LAHIR date,
ID_AGAMA int(11) NOT NULL,
FOREIGN KEY (ID_AGAMA) REFERENCES AGAMA (ID_AGAMA),
ANAK_KE int(11),
NO_KK varchar(15) NOT NULL,
FOREIGN KEY (NO_KK) REFERENCES ORANG_TUA (NO_KK),
ALAMAT varchar(250),
FOTO varchar(25) NOT NULL DEFAULT 'Pengguna.jpg',
ID_KELAS int(11) NOT NULL DEFAULT '1',
FOREIGN KEY (ID_KELAS) REFERENCES KELAS (ID_KELAS),
PASSWORD varchar(100) NOT NULL DEFAULT '123456'
);
/* 7. Tabel Guru */
CREATE TABLE GURU (
NIP varchar(25) NOT NULL PRIMARY KEY,
NAMA varchar(150),
FOTO varchar(25) NOT NULL DEFAULT 'Pengguna.jpg',
ID_KELAS int(11) NOT NULL,
FOREIGN KEY (ID_KELAS) REFERENCES KELAS (ID_KELAS),
PASSWORD varchar(100) NOT NULL DEFAULT '123456'
);

```

### 5.3.9 Implementasi Fungsi Verifikasi Pendaftaran

Fungsi verifikasi pendaftaran terdapat pada kelas *Controller* Guru, yangmana merupakan fungsi untuk memverifikasi data registrasi yang masuk. Tabel 6.4 menunjukkan bagaimana kode sumber dari fungsi verifikasi pendaftaran.

**Tabel 5.17 Implementasi Fungsi Verifikasi Pendaftaran**

public function verifikasi(\$id)
\$this->Registrasi_model->change_status(\$id);
\$registrasi = \$this->Registrasi_model->get_registrasi_by_kk(\$id)->result();
\$this->Orang_Tua_model->insert_orang_tua(\$registrasi);
\$this->Murid_model->insert_murid(\$registrasi);
redirect(base_url('Guru/registrasi_murid'));



**Gambar 5.18 Antarmuka Halaman Verifikasi Pendaftaran**

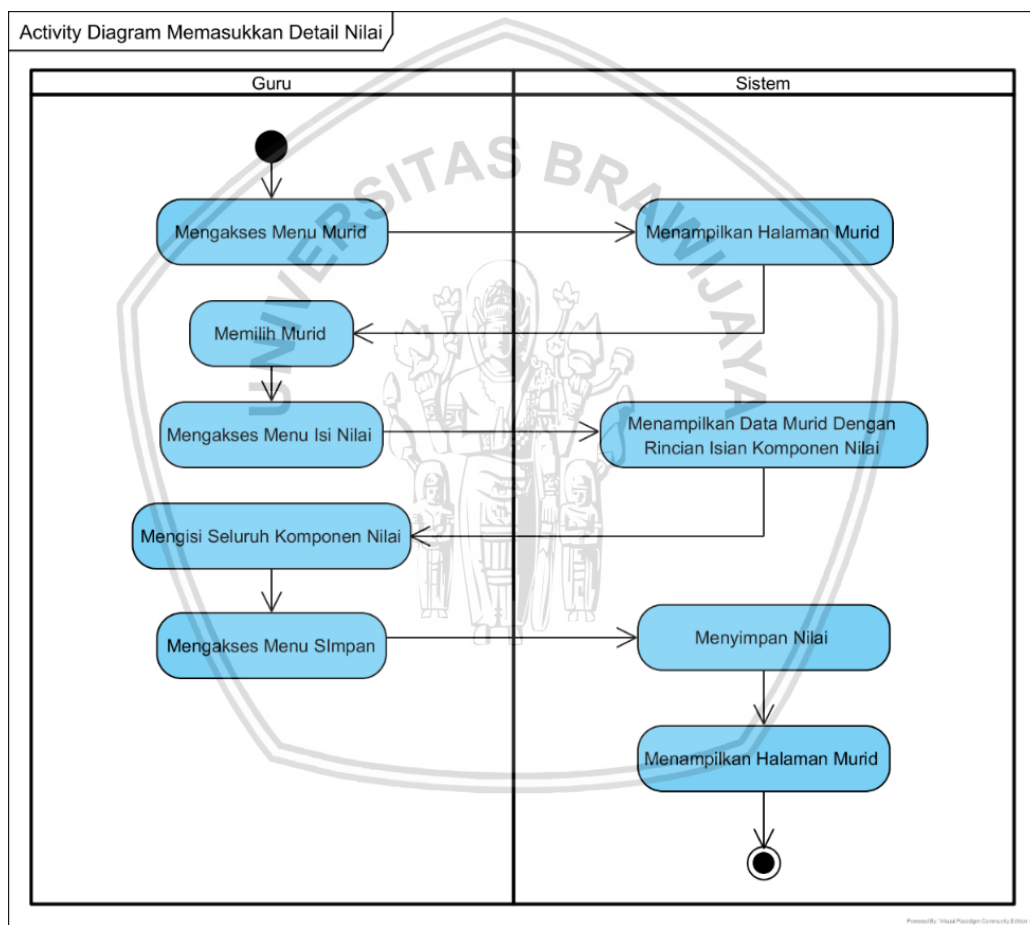
67

## 5.4 Inkremen Ketiga

Inkremen Ketiga dilakukan dengan memodelkan skenario dari *use case* kedalam *Activity Diagram* untuk mengetahui bagaimana interaksi antara pengguna dengan sistem, serta memodelkan *Sequence Diagram* untuk mengetahui bagaimana pertukaran pesan yang dilakukan antar komponen dalam sistem.

Berdasarkan Analisis Inkremental, diagram aktivitas dan diagram sekuen yang dimodelkan yaitu pada fitur Memasukkan Detail Nilai.

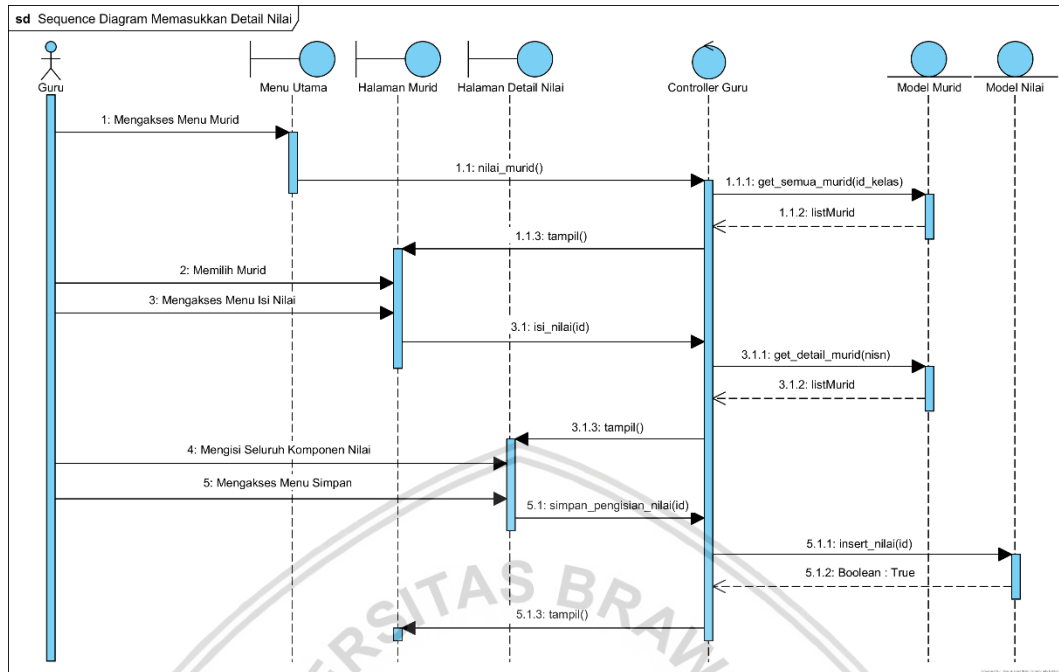
### 5.4.1 Activity Diagram Memasukkan Detail Nilai (AD-SIAS-5)



**Gambar 5.19 Activity Diagram Memasukkan Detail Nilai**

Gambar 5.17 menjelaskan alur aktivitas dimana guru melakukan pengisian detail nilai murid sesuai dengan *use case scenario* yang telah dibuat. Sehingga melalui proses ini, didapatkan detail nilai murid untuk ditampilkan pada masing-masing murid sesuai dengan NISN.

#### 5.4.2 Sequence Diagram Memasukkan Detail Nilai (SQ-SIAS-5)



**Gambar 5.20 Activity Diagram Memasukkan Detail Nilai**

Gambar 5.18 menjelaskan alur mekanisme interaksi sekuensial antar aktor guru, dengan *controller* guru yang menangani permintaan aktor guru, dengan *boundary* menu utama untuk mengakses menu, *boundary* halaman murid untuk menampilkan murid, *boundary* halaman detail nilai untuk mengisi nilai, serta *model* murid, dan *model* nilai untuk penyimpanan data.

#### 5.4.3 Pemodelan Class Diagram

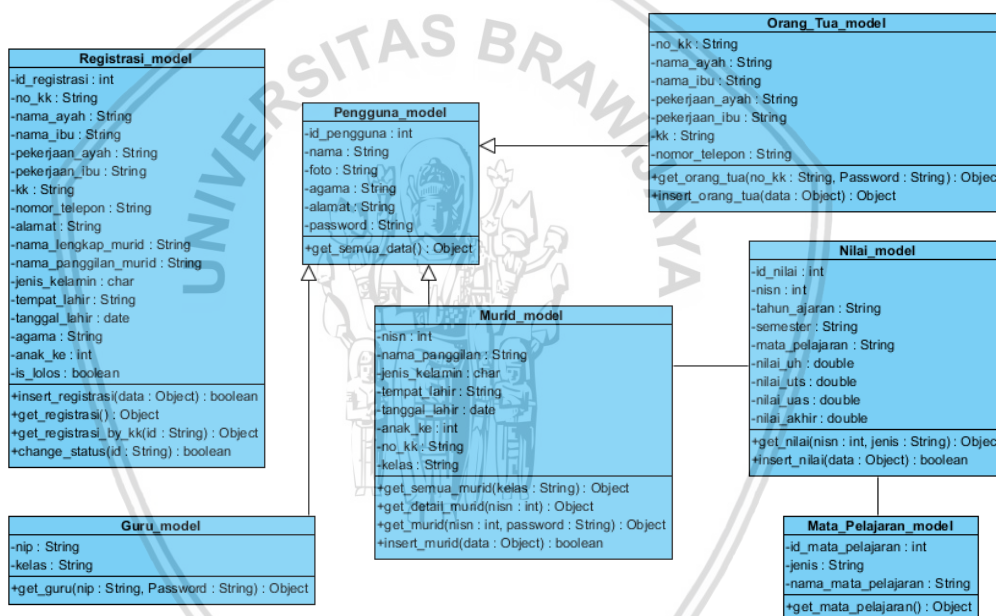
Diagram kelas dimodelkan berdasarkan diagram sekuen, yang terdapat pada fitur mengisi formulir pendaftaran murid. Dengan menggunakan prinsip *MVC* (*Model – View – Controller*), diagram kelas dapat digolongkan menjadi 2 jenis diagram kelas yakni diagram kelas sebagai *logical class* dan diagram kelas sebagai *domain model*.





**Gambar 5.21 Class Diagram Guru sebagai Logical Class**

Kelas *Logical* merupakan representasi dari pengolahan logika untuk pemanggilan kelas data, dan tampilan.

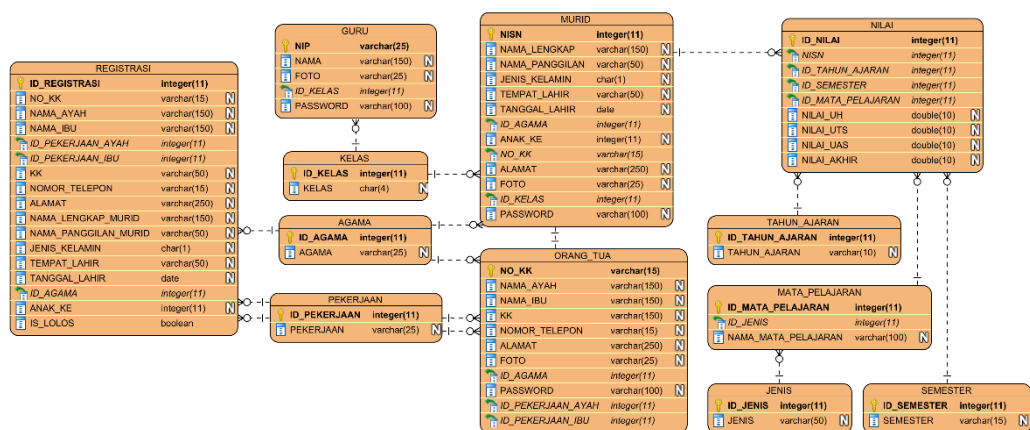


**Gambar 5.22 Class Diagram sebagai Domain Model**

Kelas Domain Model menggambarkan kelas model yang terlibat dalam proses memasukkan detail nilai dan relasi diantaranya, sehingga nantinya digunakan untuk acuan pemodelan data.

#### 5.4.4 Pemodelan Data

Data dimodelkan dalam *Physical Data Modeling*, yang dapat dilihat pada gambar 5.23



Gambar 5.23 Physical Data Modeling

Gambar 5.23 menggambarkan model tabel, yang kemudian dispesifikasikan menjadi dua tabel di bawah ini.

5.4.4.1 Tabel Agama

Tabel 5.18 Tabel Agama

No	Nama Kolom	Tipe Data	Ukuran	Constraint
1	ID_AGAMA	INTEGER	-	Primary Key
2	AGAMA	VARCHAR	25	-

5.4.4.2 Tabel Pekerjaan

Tabel 5.19 Tabel Pekerjaan

No	Nama Kolom	Tipe Data	Ukuran	Constraint
1	ID_PKERJAAN	INTEGER	-	Primary Key
2	PEKERJAAN	VARCHAR	25	-

5.4.4.3 Tabel Kelas

Tabel 5.20 Tabel Kelas

No	Nama Kolom	Tipe Data	Ukuran	Constraint
1	ID_KELAS	INTEGER	-	Primary Key
2	KELAS	CHAR	4	-

## 5.4.4.4 Tabel Registrasi

Tabel 5.21 Tabel Registrasi

No	Nama Kolom	Tipe Data	Ukuran	Constraint
1	ID_REGISTRASI	INTEGER	-	Primary Key
2	NO_KK	VARCHAR	15	-
3	NAMA_AYAH	VARCHAR	150	-
4	NAMA_IBU	VARCHAR	150	-
5	ID_PKERJAAN_AYAH	INTEGER	-	Foreign Key
6	ID_PKERJAAN_IBU	INTEGER	-	Foreign Key
7	KK	VARCHAR	50	-
8	NOMOR_TELEPON	VARCHAR	15	-
9	ALAMAT	VARCHAR	250	-
10	NAMA LENGKAP_MURID	VARCHAR	150	-
11	NAMA_PANGGILAN_MURID	VARCHAR	50	-
12	JENIS_KELAMIN	CHAR	1	-
13	TEMPAT_LAHIR	VARCHAR	50	-
14	TANGGAL_LAHIR	DATE	-	-
15	ID_AGAMA	INTEGER	-	Foreign Key
16	ANAK_KE	INTEGER	-	-
17	IS_LOLOS	BOOLEAN	-	Not Null

## 5.4.4.5 Tabel Orang Tua

Tabel 5.22 Tabel Orang Tua

No	Nama Kolom	Tipe Data	Ukuran	Constraint
1	NO_KK	VARCHAR	15	Primary Key
2	NAMA_AYAH	VARCHAR	150	-
3	NAMA_IBU	VARCHAR	150	-
4	ID_PKERJAAN_AYAH	INTEGER	-	Foreign Key
5	ID_PKERJAAN_IBU	INTEGER	-	Foreign Key
6	KK	VARCHAR	50	-
7	NOMOR_TELEPON	VARCHAR	15	-

Tabel 5.22 Tabel Orang Tua (Lanjutan)

No	Nama Kolom	Tipe Data	Ukuran	Constraint
8	ALAMAT	VARCHAR	250	-
9	FOTO	VARCHAR	25	Not Null
10	PASSWORD	VARCHAR	100	Not Null

## 5.4.4.6 Tabel Murid

Tabel 5.23 Tabel Murid

No	Nama Kolom	Tipe Data	Ukuran	Constraint
1	NISN	INTEGER	-	Primary Key
2	NAMA LENGKAP	VARCHAR	150	-
3	NAMA PANGGILAN	VARCHAR	50	-
4	JENIS KELAMIN	CHAR	1	-
5	TEMPAT LAHIR	VARCHAR	50	-
6	TANGGAL LAHIR	DATE	-	-
7	ID_AGAMA	INTEGER	-	Foreign Key
8	ANAK_KE	INTEGER	-	-
9	NO_KK	VARCHAR	15	Foreign Key
10	ALAMAT	VARCHAR	250	-
11	FOTO	VARCHAR	25	Not Null
12	ID_KELAS	INTEGER	-	Foreign Key
13	PASSWORD	VARCHAR	100	Not Null

## 5.4.4.7 Tabel Guru

Tabel 5.24 Tabel Guru

No	Nama Kolom	Tipe Data	Ukuran	Constraint
1	NIP	VARCHAR	25	Primary Key
2	NAMA	VARCHAR	150	-
3	FOTO	VARCHAR	25	Not Null
4	ID_KELAS	INTEGER	-	Foreign Key
5	PASSWORD	VARCHAR	100	Not Null

## 5.4.4.8 Tabel Tahun Ajaran

Tabel 5.25 Tabel Tahun Ajaran

No	Nama Kolom	Tipe Data	Ukuran	Constraint
1	ID_TAHUN_AJARAN	INTEGER	-	Primary Key
2	TAHUN_AJARAN	VARCHAR	10	-

## 5.4.4.9 Tabel Semester

Tabel 5.26 Tabel Semester

No	Nama Kolom	Tipe Data	Ukuran	Constraint
1	ID_SEMESTER	INTEGER	-	Primary Key
2	SEMESTER	VARCHAR	15	-

## 5.4.4.10 Tabel Jenis

Tabel 5.27 Tabel Jenis

No	Nama Kolom	Tipe Data	Ukuran	Constraint
1	ID_JENIS	INTEGER	-	Primary Key
2	JENIS	VARCHAR	50	-

## 5.4.4.11 Tabel Mata Pelajaran

Tabel 5.28 Tabel Mata Pelajaran

No	Nama Kolom	Tipe Data	Ukuran	Constraint
1	ID_MATA_PELAJARAN	INTEGER	-	Primary Key
2	ID_JENIS	INTEGER	-	Foreign Key
3	NAMA_MATA_PELAJARAN	VARCHAR	100	-

## 5.4.4.12 Tabel Nilai

Tabel 5.29 Tabel Nilai

No	Nama Kolom	Tipe Data	Ukuran	Constraint
1	ID_NILAI	INTEGER	-	Primary Key
2	NISN	INTEGER	-	Foreign Key
3	ID_TAHUN_AJARAN	INTEGER	-	Foreign Key

**Tabel 5.29 Tabel Nilai (Lanjutan)**

No	Nama Kolom	Type Data	Ukuran	Constraint
4	ID_SEMESTER	INTEGER	-	Foreign Key
5	ID_MATA_PELAJARAN	INTEGER	-	Foreign Key
6	NILAI_UH	DOUBLE	-	-
7	NILAI_UTS	DOUBLE	-	-
8	NILAI_UAS	DOUBLE	-	-
9	NILAI_AKHIR	VARCHAR	5	-

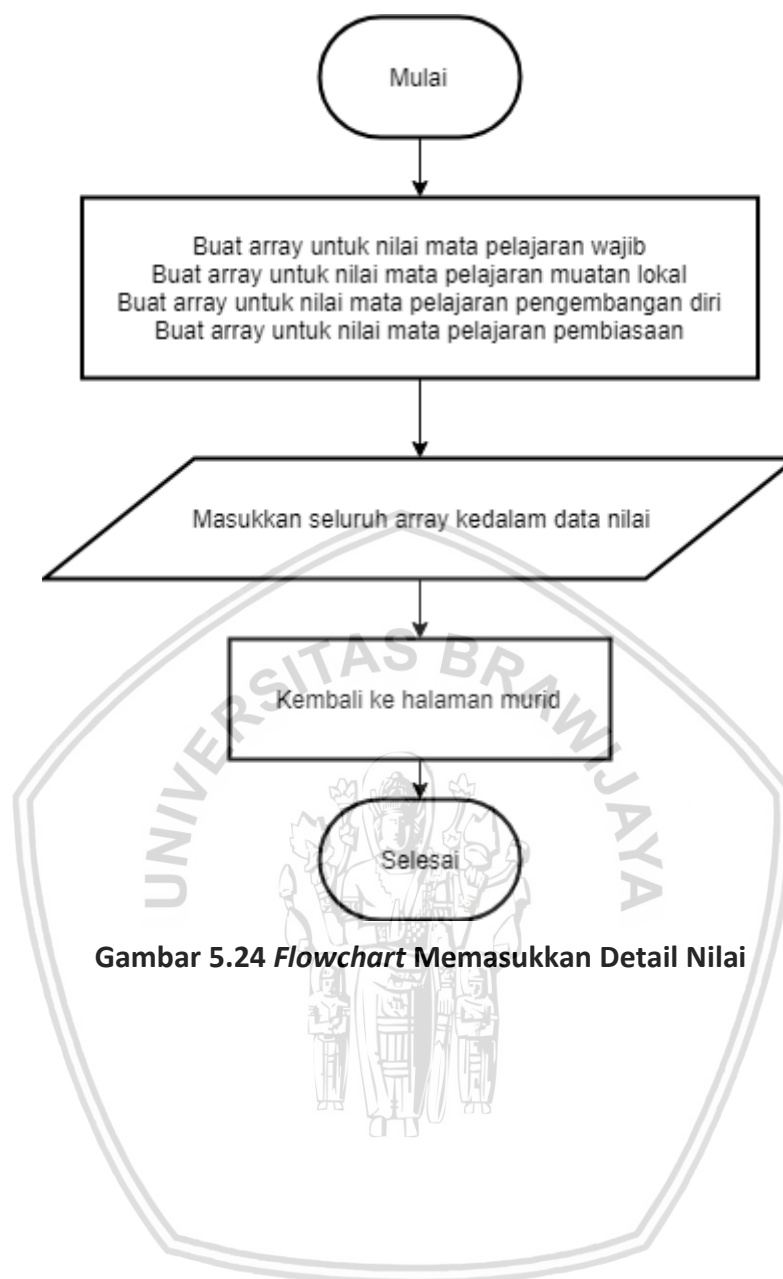
#### 5.4.5 Desain Algoritma Memasukkan Detail Nilai

Algoritma memasukkan detail nilai merupakan algoritma untuk memasukkan detail masing-masing komponen penilaian untuk masing-masing murid. Tabel 5.31 menjelaskan bagaimana urutan dari proses memasukkan detail nilai.

**Tabel 5.30 Algoritma Memasukkan Detail Nilai**

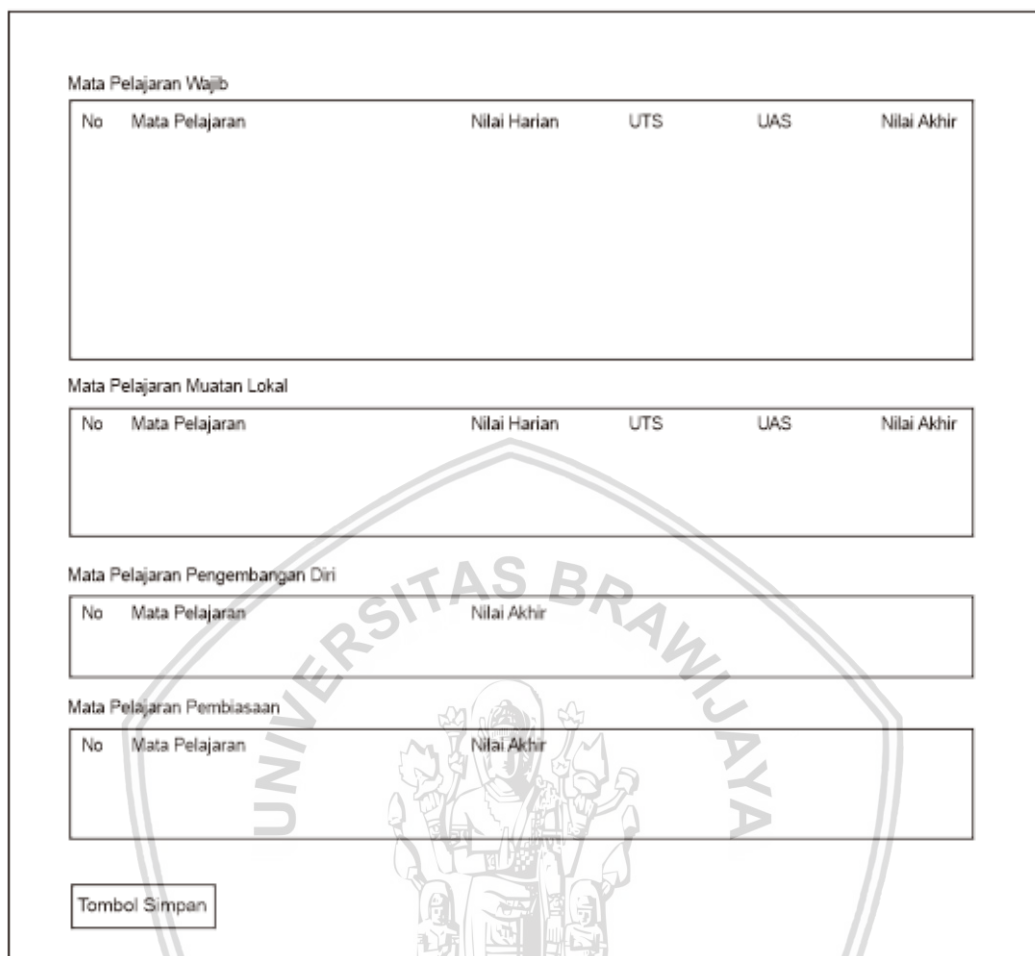
Baris	Simpan_pengisian_nilai(id)
1	Mulai
2	Buat array untuk menangkap inputan seluruh nilai mata pelajaran wajib
3	
4	Buat array untuk menangkap inputan seluruh nilai mata pelajaran muatan lokal
5	
6	Buat array untuk menangkap inputan seluruh nilai mata pelajaran pengembangan diri
7	
8	Buat array untuk menangkap inputan seluruh nilai mata pelajaran pembiasaan
9	Masukkan seluruh array kedalam data nilai
10	Kembali ke halaman murid
	Selesai





**Gambar 5.24 Flowchart Memasukkan Detail Nilai**

#### 5.4.6 Desain Antarmuka Halaman Memasukkan Detail Nilai



**Mata Pelajaran Wajib**

No	Mata Pelajaran	Nilai Harian	UTS	UAS	Nilai Akhir

**Mata Pelajaran Muatan Lokal**

No	Mata Pelajaran	Nilai Harian	UTS	UAS	Nilai Akhir

**Mata Pelajaran Pengembangan Diri**

No	Mata Pelajaran	Nilai Akhir

**Mata Pelajaran Pembiasaan**

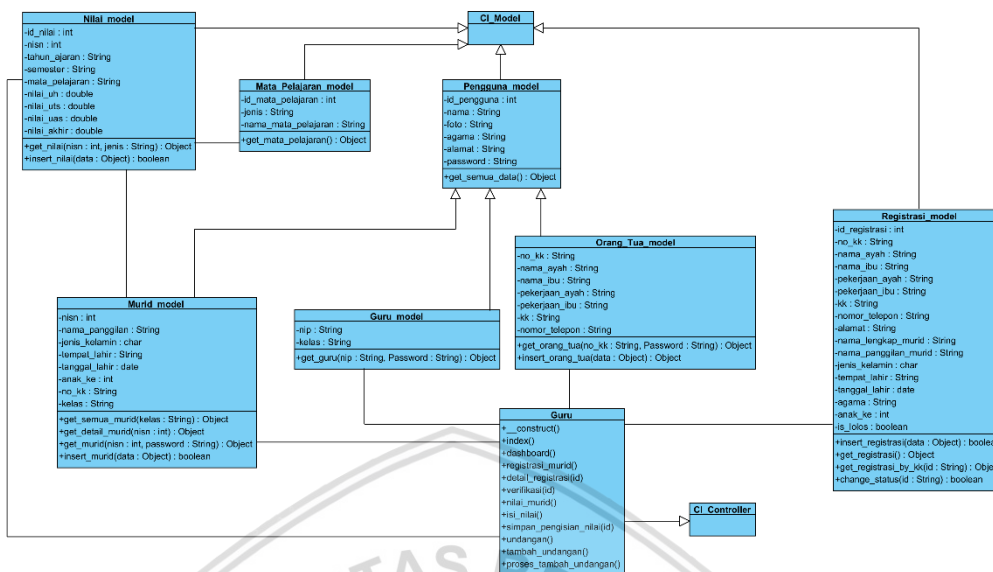
No	Mata Pelajaran	Nilai Akhir

**Tombol Simpan**

**Gambar 5.25 Desain Antarmuka Halaman Memasukkan Detail Nilai**

Desain Antarmuka Halaman memasukkan detail nilai merupakan desain antarmuka untuk memasukkan detail nilai murid. Terdapat tombol simpan untuk mengeksekusi penyimpanan nilai.

### 5.4.7 Implementasi Class Diagram



Gambar 5.26 Implementasi Class Diagram

Gambar 5.26 merupakan diagram kelas yang telah diimplementasikan menggunakan *framework CodeIgniter*.

### 5.4.8 Implementasi DDL (Data Definition Language)

Berikut merupakan implementasi DDL (*Data Definition Language*) berdasarkan rancangan, untuk penyimpanan data pada sistem, yang dapat dilihat pada tabel 5.31.

Tabel 5.31 Implementasi DDL

```

/* 1. Tabel Agama */
CREATE TABLE AGAMA (
    ID_AGAMA int(11) NOT NULL PRIMARY KEY,
    AGAMA varchar(25)
);

/* 2. Tabel Pekerjaan */
CREATE TABLE PEKERJAAN (
    ID_PKERJAAN int(11) NOT NULL PRIMARY KEY,
    PEKERJAAN varchar(25)
);

/* 3. Tabel Kelas */
CREATE TABLE KELAS (
    ID_KELAS int(11) NOT NULL PRIMARY KEY,
    KELAS char(4)
);
  
```

**Tabel 5.31 Implementasi DDL (lanjutan)**

```

/* 4. Tabel Registrasi */
CREATE TABLE REGISTRASI (
    ID_REGISTRASI int(11) NOT NULL PRIMARY KEY,
    NO_KK varchar(15),
    NAMA_AYAH varchar(150),
    NAMA_IBU varchar(150),
    ID_PKERJAAN_AYAH int(11),
    FOREIGN KEY (ID_PKERJAAN_AYAH) REFERENCES
    PEKERJAAN (ID_PKERJAAN),
    ID_PKERJAAN_IBU int(11),
    FOREIGN KEY (ID_PKERJAAN_IBU) REFERENCES
    PEKERJAAN (ID_PKERJAAN),
    KK varchar(50),
    NOMOR_TELEPON varchar(15),
    ALAMAT varchar(250),
    NAMA LENGKAP_MURID varchar(150),
    NAMA_PANGGILAN_MURID varchar(50),
    JENIS_KELAMIN char(1),
    TEMPAT_LAHIR varchar(50),
    TANGGAL_LAHIR date,
    ID_AGAMA int(11) NOT NULL,
    FOREIGN KEY (ID_AGAMA) REFERENCES AGAMA (ID_AGAMA),
    ANAK_KE int(11),
    IS_LOLOS boolean NOT NULL DEFAULT 'False'
);

/* 5. Tabel Orang Tua */
CREATE TABLE ORANG_TUA (
    NO_KK varchar(15) NOT NULL PRIMARY KEY,
    NAMA_AYAH varchar(150),
    NAMA_IBU varchar(150),
    ID_PKERJAAN_AYAH int(11),
    FOREIGN KEY (ID_PKERJAAN_AYAH) REFERENCES
    PEKERJAAN (ID_PKERJAAN),
    ID_PKERJAAN_IBU int(11),
    FOREIGN KEY (ID_PKERJAAN_IBU) REFERENCES
    PEKERJAAN (ID_PKERJAAN),
    KK varchar(50),
    NOMOR_TELEPON varchar(15),

```

**Tabel 5.31 Implementasi DDL (lanjutan)**

```

ALAMAT varchar(250),
FOTO varchar(25) NOT NULL DEFAULT 'Pengguna.jpg',
PASSWORD varchar(100) NOT NULL DEFAULT '123456'
);
/* 6. Tabel Murid */
CREATE TABLE MURID (
    NISN int(11) NOT NULL PRIMARY KEY,
    NAMA LENGKAP varchar(150),
    NAMA PANGGILAN varchar(50),
    JENIS KELAMIN char(1),
    TEMPAT LAHIR varchar(50),
    TANGGAL LAHIR date,
    ID_AGAMA int(11) NOT NULL,
    FOREIGN KEY(ID_AGAMA) REFERENCES AGAMA (ID_AGAMA),
    ANAK_KE int(11),
    NO_KK varchar(15) NOT NULL,
    FOREIGN KEY(NO_KK) REFERENCES ORANG_TUA (NO_KK),
    ALAMAT varchar(250),
    FOTO varchar(25) NOT NULL DEFAULT 'Pengguna.jpg',
    ID_KELAS int(11) NOT NULL DEFAULT '1',
    FOREIGN KEY(ID_KELAS) REFERENCES KELAS (ID_KELAS),
    PASSWORD varchar(100) NOT NULL DEFAULT '123456'
);
/* 7. Tabel Guru */
CREATE TABLE GURU (
    NIP varchar(25) NOT NULL PRIMARY KEY,
    NAMA varchar(150),
    FOTO varchar(25) NOT NULL DEFAULT 'Pengguna.jpg',
    ID_KELAS int(11) NOT NULL,
    FOREIGN KEY(ID_KELAS) REFERENCES KELAS (ID_KELAS),
    PASSWORD varchar(100) NOT NULL DEFAULT '123456'
);
/* 8. Tabel Tahun Ajaran */
CREATE TABLE TAHUN_AJARAN (
    ID_TAHUN_AJARAN int(11) NOT NULL PRIMARY KEY,
    TAHUN_AJARAN varchar(10)
);

```

**Tabel 5.31 Implementasi DDL (lanjutan)**

```

/* 9. Tabel Semester */
CREATE TABLE SEMESTER (
    ID_SEMESTER int(11) NOT NULL PRIMARY KEY,
    SEMESTER varchar(15)
);

/* 10. Tabel Jenis */
CREATE TABLE JENIS (
    ID_JENIS int(11) NOT NULL PRIMARY KEY,
    JENIS varchar(50)
);

/* 11. Tabel Mata_Pelajaran */
CREATE TABLE MATA_PELAJARAN (
    ID_MATA_PELAJARAN int(11) NOT NULL PRIMARY KEY,
    ID_JENIS int(11) NOT NULL,
    FOREIGN KEY(ID_JENIS) REFERENCES JENIS (ID_JENIS),
    NAMA_MATA_PELAJARAN varchar(100)
);

/* 12. Tabel Nilai */
CREATE TABLE NILAI (
    ID_NILAI int(11) NOT NULL PRIMARY KEY,
    NISN int(11) NOT NULL,
    FOREIGN KEY(NISN) REFERENCES MURID (NISN),
    ID_TAHUN_AJARAN int(11) NOT NULL,
    FOREIGN KEY(ID_TAHUN_AJARAN) REFERENCES TAHUN_AJARAN
(ID_TAHUN_AJARAN),
    ID_SEMESTER int(11) NOT NULL,
    FOREIGN KEY(ID_SEMESTER) REFERENCES SEMESTER (ID_SEMESTER),
    ID_MATA_PELAJARAN int(11) NOT NULL,
    FOREIGN KEY(ID_MATA_PELAJARAN) REFERENCES MATA_PELAJARAN
(ID_MATA_PELAJARAN),
    NILAI_UH double,
    NILAI_UTS double,
    NILAI_UAS double,
    NILAI_AKHIR varchar(5)
);

```



#### 5.4.9 Implementasi Fungsi Memasukkan Detail Nilai

Fungsi autentikasi terdapat pada kelas *Controller* Guru, yangmana merupakan fungsi untuk memasukkan nilai murid untuk masing-masing komponen penilaian dan masing-masing murid. Tabel 5.32 menunjukkan bagaimana kode sumber dari fungsi memasukkan detail nilai.

**Tabel 5.32 Implementasi Fungsi Memasukkan Detail Nilai**

<pre> public function simpan_pengisian_nilai(\$id)  \$wajib1 = array(     'nisn' =&gt; \$id,     'id_mata_pelajaran' =&gt; 1,     'nilai_uh' =&gt; \$this-&gt;input-&gt;post('Aga_UH'),     'nilai_uts' =&gt; \$this-&gt;input-&gt;post('Aga_UTS'),     'nilai_uas' =&gt; \$this-&gt;input-&gt;post('Aga_UAS'),     'nilai_akhir' =&gt; \$this-&gt;input-&gt;post('Aga_Akhir')); \$wajib2 = array(     'nisn' =&gt; \$id,     'id_mata_pelajaran' =&gt; 2,     'nilai_uh' =&gt; \$this-&gt;input-&gt;post('Pkn_UH'),     'nilai_uts' =&gt; \$this-&gt;input-&gt;post('Pkn_UTS'),     'nilai_uas' =&gt; \$this-&gt;input-&gt;post('Pkn_UAS'),     'nilai_akhir' =&gt; \$this-&gt;input-&gt;post('Pkn_Akhir')); \$wajib3 = array(     'nisn' =&gt; \$id,     'id_mata_pelajaran' =&gt; 3,     'nilai_uh' =&gt; \$this-&gt;input-&gt;post('BIndo_UH'),     'nilai_uts' =&gt; \$this-&gt;input-&gt;post('BIndo_UTS'),     'nilai_uas' =&gt; \$this-&gt;input-&gt;post('BIndo_UAS'),     'nilai_akhir' =&gt; \$this-&gt;input-&gt;post('BIndo_Akhir')); \$wajib4 = array(     'nisn' =&gt; \$id,     'id_mata_pelajaran' =&gt; 4,     'nilai_uh' =&gt; \$this-&gt;input-&gt;post('Mat_UH'),     'nilai_uts' =&gt; \$this-&gt;input-&gt;post('Mat_UTS'),     'nilai_uas' =&gt; \$this-&gt;input-&gt;post('Mat_UAS'),     'nilai_akhir' =&gt; \$this-&gt;input-&gt;post('Mat_Akhir')); \$wajib5 = array(     'nisn' =&gt; \$id,     'id_mata_pelajaran' =&gt; 5, </pre>
--

**Tabel 5.32 Implementasi Fungsi Memasukkan Detail Nilai (Lanjutan)**

```

public function simpan_pengisian_nilai($id)

    'nilai_uh' => $this->input->post('IPA_UH'),
    'nilai_uts' => $this->input->post('IPA_UTS'),
    'nilai_uas' => $this->input->post('IPA_UAS'),
    'nilai_akhir' => $this->input->post('IPA_Akhir'));
$wajib6 = array(
    'nisp' => $id,
    'id_mata_pelajaran' => 6,
    'nilai_uh' => $this->input->post('IPS_UH'),
    'nilai_uts' => $this->input->post('IPS_UTS'),
    'nilai_uas' => $this->input->post('IPS_UAS'),
    'nilai_akhir' => $this->input->post('IPS_Akhir'));
$wajib7 = array(
    'nisp' => $id,
    'id_mata_pelajaran' => 7,
    'nilai_uh' => $this->input->post('SBK_UH'),
    'nilai_uts' => $this->input->post('SBK_UTS'),
    'nilai_uas' => $this->input->post('SBK_UAS'),
    'nilai_akhir' => $this->input->post('SBK_Akhir'));
$wajib8 = array(
    'nisp' => $id,
    'id_mata_pelajaran' => 8,
    'nilai_uh' => $this->input->post('PJOK_UH'),
    'nilai_uts' => $this->input->post('PJOK_UTS'),
    'nilai_uas' => $this->input->post('PJOK_UAS'),
    'nilai_akhir' => $this->input->post('PJOK_Akhir'));
$mulok = array(
    'nisp' => $id,
    'id_mata_pelajaran' => 9,
    'nilai_uh' => $this->input->post('BJawa_UH'),
    'nilai_uts' => $this->input->post('BJawa_UTS'),
    'nilai_uas' => $this->input->post('BJawa_UAS'),
    'nilai_akhir' => $this->input->post('BJawa_Akhir'));
$pd1 = array(
    'nisp' => $id,
    'id_mata_pelajaran' => 10,

```

**Tabel 5.32 Implementasi Fungsi Memasukkan Detail Nilai (Lanjutan)**

```

public function simpan_pengisian_nilai($id)

    'nilai_uh' => NULL,
    'nilai_uts' => NULL,
    'nilai_uas' => NULL,
    'nilai_akhir' => $this->input->post('BInggris_Akhir'));
$pd2 = array(
    'nisn' => $id,
    'id_mata_pelajaran' => $this->input-
>post('Pengembangan_Diri'),
    'nilai_uh' => NULL,
    'nilai_uts' => NULL,
    'nilai_uas' => NULL,
    'nilai_akhir' => $this->input->post('PD_Akhir'));
$pembiasaan1 = array(
    'nisn' => $id,
    'id_mata_pelajaran' => 17,
    'nilai_uh' => NULL,
    'nilai_uts' => NULL,
    'nilai_uas' => NULL,
    'nilai_akhir' => $this->input->post('Disiplin_Akhir'));
$pembiasaan2 = array(
    'nisn' => $id,
    'id_mata_pelajaran' => 18,
    'nilai_uh' => NULL,
    'nilai_uts' => NULL,
    'nilai_uas' => NULL,
    'nilai_akhir' => $this->input->post('Kerjasama_Akhir'));
$pembiasaan3 = array(
    'nisn' => $id,
    'id_mata_pelajaran' => 19,
    'nilai_uh' => NULL,
    'nilai_uts' => NULL,
    'nilai_uas' => NULL,
    'nilai_akhir' => $this->input->post('Mandiri_Akhir')
);
$this->Nilai_model->insert_nilai($wajib1);
$this->Nilai_model->insert_nilai($wajib2);

```

**Tabel 5.32 Implementasi Fungsi Memasukkan Detail Nilai (Lanjutan)**

<pre> public function simpan_pengisian_nilai(\$id) \$this-&gt;Nilai_model-&gt;insert_nilai(\$wajib3); \$this-&gt;Nilai_model-&gt;insert_nilai(\$wajib4); \$this-&gt;Nilai_model-&gt;insert_nilai(\$wajib5); \$this-&gt;Nilai_model-&gt;insert_nilai(\$wajib6); \$this-&gt;Nilai_model-&gt;insert_nilai(\$wajib7); \$this-&gt;Nilai_model-&gt;insert_nilai(\$wajib8); \$this-&gt;Nilai_model-&gt;insert_nilai(\$mulok); \$this-&gt;Nilai_model-&gt;insert_nilai(\$pd1); \$this-&gt;Nilai_model-&gt;insert_nilai(\$pd2); \$this-&gt;Nilai_model-&gt;insert_nilai(\$pembiasaan1); \$this-&gt;Nilai_model-&gt;insert_nilai(\$pembiasaan2); \$this-&gt;Nilai_model-&gt;insert_nilai(\$pembiasaan3); redirect(base_url('Guru/nilai_murid')); </pre>
---



#### 5.4.10 Antarmuka Halaman Memasukkan Detail Nilai

ere on the bookmarks bar. [Import bookmarks now...](#)

No	Mata Pelajaran	Ulangan Harian	UTS	UAS	Nilai Akhir
1	Pendidikan Agama	82	90	90	88
2	Pendidikan Kewarganegaraan	80	91	91	88
3	Bahasa Indonesia	87	92	94	91
4	Matematika	91	87	88	83
5	Ilmu Pengetahuan Alam	82	79	78	81
6	Ilmu Pengetahuan Sosial	94	91	93	92
7	Seni Budaya dan Kerajinan	91	89	70	83
8	Pendidikan Jasmani Olahraga dan Kesehatan	90	84	94	89

**Mata Pelajaran Muatan Lokal**

No	Mata Pelajaran	Ulangan Harian	UTS	UAS	Nilai Akhir
1	Bahasa Jawa	85	90	95	90

**Mata Pelajaran Pengembangan Diri**

No	Mata Pelajaran	Nilai Akhir
1	Bahasa Inggris	A
2	Seni Tari	B+

**Mata Pelajaran Pembiasaan**

No	Mata Pelajaran	Nilai Akhir
1	Kedisiplinan dan Tanggungjawab	A
2	Kerjasama	B
3	Kemandirian	B+

Klik Simpan untuk menyimpan pengisian

**Simpan**

**Gambar 5.27 Antarmuka Halaman Memasukkan Detail Nilai**

Halaman memasukkan detail nilai merupakan untuk memasukkan detail nilai murid. Terdapat tombol simpan untuk mengeksekusi penyimpanan nilai.

## BAB 6 PENGUJIAN

Pengujian dilakukan untuk memastikan sistem informasi yang dibangun berfungsi dengan baik dan dapat memenuhi kebutuhan *stakeholder* terkait. Ada beberapa metode pengujian yang akan dijalankan, yakni pengujian *black-box validation testing* untuk menguji persyaratan fungsional sistem, serta pengujian *User Acceptance Testing* untuk menguji penerimaan sistem terhadap stakeholder.

### 6.1 Pengujian Black Box – Validation Testing

Pengujian ini dilakukan untuk memastikan fungsionalitas sistem. Pengujian dilakukan dengan cara mendefinisikan kasus uji berdasarkan scenario use case, dan dilakukan perbandingan antara hasil dengan ekspektasi pengujian.

#### 6.1.1 Pengujian Fitur Mengisi Formulir Pendaftaran Murid

Pengujian fitur mengisi formulir pendaftaran murid dilakukan dengan kasus uji pada *basic flow* dan alt1.

**Tabel 6.1 Kasus uji mengisi formulir pendaftaran murid: *basic flow***

<b>Test Case Code</b>	BB-001
<b>Test Case Name</b>	Test Case mengisi formulir pendaftaran murid
<b>Code Use Case</b>	UC-SIAS-1
<b>Testing Objectives</b>	Pengujian ini dilakukan untuk memastikan orang tua dapat mendaftarkan murid
<b>Test Case</b>	Memasukkan semua isian yang ada beserta lampiran gambar yang harus diunggah
<b>Testing Step</b>	<ol style="list-style-type: none"> <li>1. Penguji mengakses menu registrasi</li> <li>2. Penguji mengisi seluruh isian formulir dan menyertakan berkas yang perlu diunggah</li> <li>3. Penguji menekan tombol daftar</li> </ol>
<b>Expectation</b>	Menampilkan halaman registrasi
<b>Result</b>	Sistem menampilkan halaman registrasi
<b>Status</b>	Valid

Tabel 6.1 berisi kasus uji *black box* dengan kode pengujian BB-001. Kasus uji dilakukan dengan memasukkan semua data isian yang ada, beserta melampirkan gambar yang harus diunggah. *Result* dari pengujian menunjukkan hasil yang valid. Gambar 6.1, gambar 6.2, dan gambar 6.3 merupakan hasil dari pengujian pada kasus ini.



**FORMULIR PENDAFTARAN MURID BARU**

Batas Pendaftaran Hingga April 7, 2018

Nomor KK  
753812638

Nama Ayah  
Ayah

Nama Ibu  
Ibu

Pekerjaan Ayah  
Pegawai Negeri

Pekerjaan Ibu  
Swasta

Nomor Telepon  
09798786

**Gambar 6.1 Kasus uji mengisi formulir pendaftaran murid: *basic flow***

Tempat Lahir  
Kediri

Tanggal Lahir  
04/10/2018

Agama  
Islam

Anak Ke  
1

Unggah Foto KK (Ukuran Maksimal 2 MB)  
Choose File chart (1).png

Daftar

**Gambar 6.2 Kasus uji mengisi formulir pendaftaran murid: *basic flow***

**Gambar 6.3 Kasus uji mengisi formulir pendaftaran murid: *basic flow***

**Tabel 6.2 Kasus uji mengisi formulir pendaftaran murid: *alt1***

<b>Test Case Code</b>	BB-002
<b>Test Case Name</b>	Test Case mengisi formulir pendaftaran murid
<b>Code Use Case</b>	UC-SIAS-1
<b>Testing Objectives</b>	Pengujian ini dilakukan untuk memastikan orang tua tidak dapat mendaftarkan murid jika data yang diisi tidak lengkap
<b>Test Case</b>	Memasukkan sebagian dari isian yang ada dan tidak menyertakan lampiran gambar yang harus diunggah
<b>Testing Step</b>	<ol style="list-style-type: none"> <li>1. Penguji mengakses menu registrasi</li> <li>2. Penguji mengisi sebagian isian formulir dan menyertakan berkas yang perlu diunggah</li> <li>3. Penguji menekan tombol daftar</li> </ol>
<b>Expectation</b>	Menampilkan pesan isi bagian formulir yang kosong
<b>Result</b>	Sistem menampilkan pesan isi bagian formulir yang kosong
<b>Status</b>	Valid

Tabel 6.2 berisi kasus uji *black box* dengan kode pengujian BB-002. Kasus uji dilakukan dengan memasukkan sebagian dari isian yang ada dan tidak menyertakan lampiran gambar yang harus diunggah. *Result* dari pengujian menunjukkan hasil yang valid. Gambar 6.4, gambar 6.5, dan gambar 6.6 merupakan hasil dari pengujian pada kasus ini.

HOME REGISTRASI ALUMNI LOGIN

### FORMULIR PENDAFTARAN MURID BARU

Batas Pendaftaran Hingga April 7, 2018

Nomor KK  
1415113

Nama Ayah  
Ayah

Nama Ibu  
Ibu

Pekerjaan Ayah

Pekerjaan Ibu

Nomor Telepon

**Gambar 6.4 Kasus uji mengisi formulir pendaftaran murid: alt1**

Tempat Lahir

Tanggal Lahir  
mm/dd/yyyy

Agama  
Islam

Anak Ke

Unggah Foto KK (Ukuran Maksimal 2 MB)  
Choose File No file chosen

Daftar

**Gambar 6.5 Kasus uji mengisi formulir pendaftaran murid: alt1**

HOME REGISTRASI ALUMNI LOGIN

Nama Ayah  
Ayah

Nama Ibu  
Ibu

Pekerjaan Ayah

Pekerjaan Ibu  
Please fill out this field.

Nomor Telepon

Alamat

Nama Lengkap Calon Murid

**Gambar 6.6 Kasus uji mengisi formulir pendaftaran murid: alt1**

### 6.1.2 Pengujian Fitur Melakukan Verifikasi Pendaftaran

Pengujian fitur melakukan verifikasi pendaftaran dilakukan dengan kasus uji pada *basic flow* pada use case melakukan verifikasi pendaftaran.

**Tabel 6.3 Kasus uji melakukan verifikasi pendaftaran: *basic flow***

<b>Test Case Code</b>	BB-003
<b>Test Case Name</b>	Test Case melakukan verifikasi pendaftaran
<b>Code Use Case</b>	UC-SIAS-2
<b>Testing Objectives</b>	Pengujian ini dilakukan untuk memastikan guru dapat memverifikasi pendaftaran murid
<b>Test Case</b>	Memilih tombol verifikasi yang terdapat pada halaman verifikasi
<b>Testing Step</b>	<ol style="list-style-type: none"> <li>1. Penguji memilih menu registrasi</li> <li>2. Penguji memilih salah satu data registrasi murid</li> <li>3. Penguji menekan tombol verifikasi</li> </ol>
<b>Expectation</b>	Mengubah status pendaftar menjadi terverifikasi
<b>Result</b>	Sistem mengubah status pendaftar menjadi terverifikasi
<b>Status</b>	Valid

Tabel 6.3 berisi kasus uji *black box* dengan kode pengujian BB-003. Kasus uji dilakukan dengan menekan tombol verifikasi yang terdapat pada halaman verifikasi. *Result* dari pengujian ini menunjukkan hasil yang valid. Gambar 6.7, gambar 6.8, gambar 6.9, dan gambar 6.10 merupakan hasil dari pengujian pada kasus ini.

No	Nomor KK	Nama Lengkap	Nama Ayah	Nama Ibu	Nomor Telepon	Detail Verifikasi
1	14045	Tatjana Saphira Hartmann	Joachim Hartmann	Ade Hartmann	081234567890	✓ Sudah Diverifikasi
2	3506060612180	Faraday Al Jawara	Jawara	Wahyu	089612481	✓ Sudah Diverifikasi
3	13124124	a	a	a	8712	Tampilkan
4	123455465	gggg	aaaaa	bbbbb	08123	Tampilkan
5	753812638	Murid	Ayah	Ibu	09798786	Tampilkan

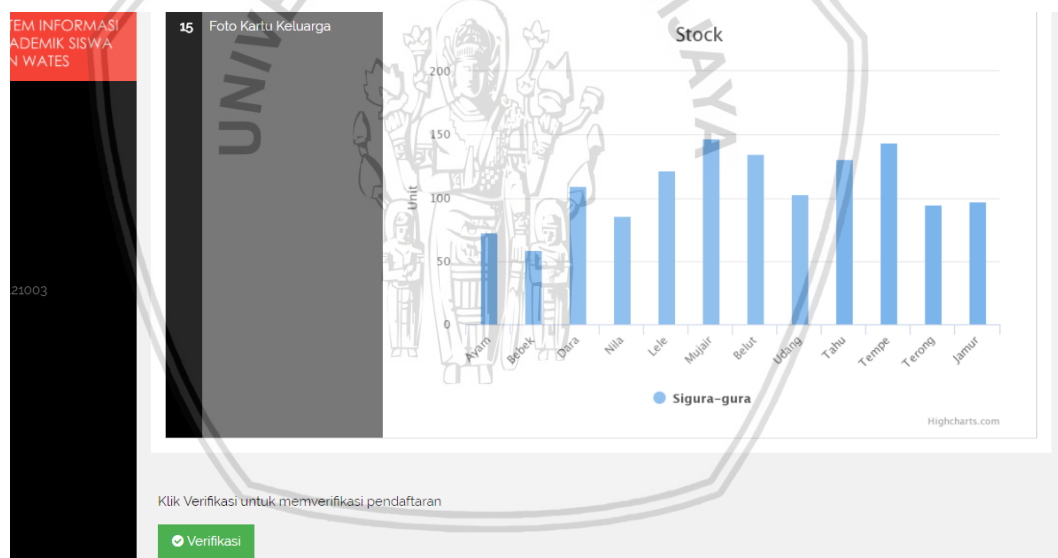
**Gambar 6.7 Kasus uji melakukan verifikasi pendaftaran: *basic flow***

## Detail Registrasi

[Registrasi >> Detail Registrasi](#)

1	ID Registrasi	5
2	Nomor Kartu Keluarga	753812638
3	Nama Ayah	Ayah
4	Nama Ibu	Ibu
5	Pekerjaan Ayah	Pegawai Negeri
6	Pekerjaan Ibu	Swasta
7	Nomor Telepon	09798786
8	Nama Lengkap Murid	Murid
9	Nama Panggilan	Murid
10	Jenis Kelamin	L
11	Tempat, dan Tanggal Lahir	Kediri, 2018-04-10

**Gambar 6.8 Kasus uji melakukan verifikasi pendaftaran: *basic flow***



**Gambar 6.9 Kasus uji melakukan verifikasi pendaftaran: *basic flow***

STEM INFORMASI KADEMIK SISWA ON WATES						
Registrasi Murid Baru						
No	Nomor KK	Nama Lengkap	Nama Ayah	Nama Ibu	Nomor Telepon	Detail Verifikasi
1	14045	Tatjana Saphira Hartmann	Joachim Hartmann	Ade Hartmann	081234567890	✓ Sudah Diverifikasi
2	3506060612180	Faraday Al Jawara	Jawara	Wahyu	089612481	✓ Sudah Diverifikasi
3	13124124	a	a	a	8712	Tampilkan
4	123455465	gggg	aaaaa	bbbbb	08123	Tampilkan
5	753812638	Murid	Ayah	Ibu	09798786	✓ Sudah Diverifikasi

Gambar 6.10 Kasus uji melakukan verifikasi pendaftaran: *basic flow*

### 6.1.3 Pengujian Fitur Memasukkan Detail Nilai

Pengujian fitur memasukkan detail nilai dilakukan dengan kasus uji pada *basic flow* pada use case memasukkan detail nilai.

Tabel 6.4 Kasus uji memasukkan detail nilai: *basic flow*

<b>Test Case Code</b>	BB-004
<b>Test Case Name</b>	Test Case memasukkan detail nilai
<b>Code Use Case</b>	UC-SIAS-5
<b>Testing Objectives</b>	Pengujian ini dilakukan untuk memastikan guru dapat memasukkan detail nilai
<b>Test Case</b>	Mengisi seluruh komponen nilai
<b>Testing Step</b>	<ol style="list-style-type: none"> <li>1. Penguji memilih menu murid</li> <li>2. Penguji memilih salah satu data murid</li> <li>3. Penguji mengisi seluruh komponen penilaian</li> <li>4. Penguji menekan tombol simpan</li> </ol>
<b>Expectation</b>	Nilai murid yang dituju telah terisi, dan mengubah status menjadi sudah diisi
<b>Result</b>	Sistem menyimpan nilai murid, dengan menampilkan status isi nilai menjadi sudah terisi
<b>Status</b>	Valid

Tabel 6.4 berisi kasus uji *black box* dengan kode pengujian BB-004. Kasus uji dilakukan dengan memasukkan seluruh komponen nilai yang ada. *Result* dari pengujian menunjukkan hasil yang valid. Gambar 6.11, gambar 6.12, gambar 6.13, dan gambar 6.14 merupakan hasil dari pengujian pada kasus ini.



INFORMASI  
MIK SISWA  
TES

## Daftar Murid

Kelas **1 A**

No	NISN	Nama	Jenis Kelamin	Isi Nilai
1	3281	Gigih Caesar Wahyu Pratama	L	✓ Sudah Diisi
2	3282	Mahendra A	L	✓ Sudah Diisi
3	3283	Seto Junior	L	✓ Sudah Diisi
4	3284	Faraday Al Jawara	L	<input type="text"/>
5	3285	Tatjana Saphira Hartmann	P	<input type="text"/>
6	3286	Murid	L	<input type="text"/>

Gambar 6.11 Kasus uji memasukkan detail nilai: *basic flow*

SIAS  
SDNW

SISTEM INFORMASI  
AKADEMIK SISWA  
SDN WATES

**Lilik Purwanti**  
NIP : 198503302003121003  
Guru Kelas : 1 A

- DASBOR
- REGISTRASI
- MURID
- UNDANGAN
- LOGOUT

## Isi Nilai

Nama : Faraday Al Jawara  
NISN : 3284

Mata Pelajaran Wajib

No	Mata Pelajaran	Ulangan Harian	UTS	UAS	Nilai Akhir
1	Pendidikan Agama	80	80	80	80
2	Pendidikan Kewarganegaraan	80	80	80	80
3	Bahasa Indonesia	80	80	80	80
4	Matematika	80	80	80	80
5	Ilmu Pengetahuan Alam	80	80	80	80

Gambar 6.12 Kasus uji memasukkan detail nilai: *basic flow*

**Lilik Purwanti**  
NIP : 198503302003121003  
Guru Kelas : 1 A

- DASBOR
- REGISTRASI
- MURID
- UNDANGAN
- LOGOUT

## Mata Pelajaran Pembiasaan

No	Mata Pelajaran	Nilai Akhir
1	Kedisiplinan dan Tanggungjawab	A
2	Kerjasama	A
3	Kemandirian	A

Klik Simpan untuk menyimpan pengisian

Gambar 6.13 Kasus uji memasukkan detail nilai: *basic flow*

**SIAS SDNW** | SISTEM INFORMASI AKADEMIK SISWA SDN WATES

**Daftar Murid**

Kelas **1 A**

No	NISN	Nama	Jenis Kelamin	Isi Nilai
1	3281	Gigih Caesar Wahyu Pratama	L	✓ Sudah Diisi
2	3282	Mahendra A	L	✓ Sudah Diisi
3	3283	Seto Junior	L	✓ Sudah Diisi
4	3284	Faraday Al Jawara	L	✓ Sudah Diisi
5	3285	Tatjana Saphira Hartmann	P	
6	3286	Murid	L	

Gambar 6.14 Kasus uji memasukkan detail nilai: *basic flow*

#### 6.1.4 Hasil Pengujian *Black Box – Validation Testing*

Pengujian dengan menggunakan metode *black box – validation testing* diberlakukan pada beberapa fitur dalam sistem informasi manajemen akademik sekolah dasar. Persentase validitas pengujian dapat dihitung menggunakan cara berikut:

$$\text{Persentase} = (\text{Jumlah Uji Kasus Valid} / \text{Jumlah Uji Kasus Diujikan}) \times 100\%$$

$$\text{Persentase} = (4 / 4) \times 100\%$$

$$\text{Persentase} = 100\% \text{ (Valid)}$$

#### 6.2 Pengujian *User Acceptance Testing*

Pengujian ini bertujuan untuk memastikan sistem informasi yang dibangun dapat diterima oleh pengguna, serta dapat menyelesaikan masalah yang telah dirumuskan. Pada pengujian *User Acceptance Testing*, Tabel 6.5 menunjukkan hasil yang telah dirangkum dari masing-masing responden.

Tabel 6.5 Pengujian UAT

No	Kebutuhan Penerimaan	Penilaian				
		Sangat Baik	Baik	Netral	Kurang	Sangat Kurang
1	Mengisi Formulir Pendaftaran Murid	1	3	0	0	0
2	Melakukan Verifikasi Pendaftaran	0	2	0	0	0
3	Memasukkan Detail Nilai	2	2	0	0	0

### 6.2.1 Hasil Pengujian *User Acceptance Testing*

Hasil dari masing-masing poin kebutuhan penerimaan dianalisis menggunakan skala *likert*, dan diuraikan pada beberapa poin berikut:

1. *User Acceptance Testing* mengisi formulir pendaftaran murid
  - a. Total nilai berdasarkan jawaban dari 4 responden (orang tua calon murid)
 
$$A = 1 \times 5 = 5$$

$$B = 3 \times 4 = 12$$

$$C = 0 \times 3 = 0$$

$$D = 0 \times 2 = 0$$

$$E = 0 \times 1 = 0$$

$$\text{Total Nilai} = A + B + C + D + E$$

$$\text{Total Nilai} = 5 + 12 + 0 + 0 + 0 = 17$$
  - b. Nilai Y
 
$$Y = 5 \times 4 \times 1 = 20$$
  - c. Nilai Persentase  
 Nilai Y digunakan sebagai pembagi karena pertanyaan yang diajukan berupa pertanyaan positif.  

$$\text{Rumus Index} = (\text{Total Nilai} / Y) \times 100\%$$

$$\text{Rumus Index} = (17 / 20) \times 100\% = 85\% \text{ (Sangat Setuju)}$$
 Sehingga dapat disimpulkan bahwa orang tua sangat setuju dengan adanya fitur ini.
2. *User Acceptance Testing* melakukan verifikasi pendaftaran
  - a. Total nilai berdasarkan jawaban dari 2 responden (guru)
 
$$A = 0 \times 5 = 0$$

$$B = 2 \times 4 = 8$$

$$C = 0 \times 3 = 0$$

$$D = 0 \times 2 = 0$$

$$E = 0 \times 1 = 0$$

$$\text{Total Nilai} = A + B + C + D + E$$

$$\text{Total Nilai} = 0 + 8 + 0 + 0 + 0 = 8$$
  - b. Nilai Y
 
$$Y = 5 \times 2 \times 1 = 10$$
  - c. Nilai Persentase

Nilai Y digunakan sebagai pembagi karena pertanyaan yang diajukan berupa pertanyaan positif.

$$\text{Rumus Index} = (\text{Total Nilai} / Y) \times 100\%$$

$$\text{Rumus Index} = (8 / 10) \times 100\% = 80\% \text{ (Sangat Setuju)}$$

Sehingga dapat disimpulkan bahwa guru sangat setuju dengan adanya fitur ini.

3. *User Acceptance Testing* memasukkan detail nilai

- a. Total nilai berdasarkan jawaban dari 4 responden (guru)

$$A = 2 \times 5 = 10$$

$$B = 2 \times 4 = 8$$

$$C = 0 \times 3 = 0$$

$$D = 0 \times 2 = 0$$

$$E = 0 \times 1 = 0$$

$$\text{Total Nilai} = A + B + C + D + E$$

$$\text{Total Nilai} = 10 + 8 + 0 + 0 + 0 = 18$$

- b. Nilai Y

$$Y = 5 \times 4 \times 1 = 20$$

- c. Nilai Persentase

Nilai Y digunakan sebagai pembagi karena pertanyaan yang diajukan berupa pertanyaan positif.

$$\text{Rumus Index} = (\text{Total Nilai} / Y) \times 100\%$$

$$\text{Rumus Index} = (18 / 20) \times 100\% = 90\% \text{ (Sangat Setuju)}$$

Sehingga dapat disimpulkan bahwa guru sangat setuju dengan adanya fitur ini.

## BAB 7 KESIMPULAN

### 7.1 Kesimpulan

Berdasarkan penelitian yang dilakukan, didapatkan kesimpulan sebagai berikut:

1. Proses pendaftaran dan seleksi murid dapat dipermudah dengan menggunakan Sistem Informasi Manajemen Akademik Sekolah Dasar Negeri Wates Kabupaten Kediri, berdasarkan hasil pengujian *User Acceptance Testing* yang diberikan kepada pihak orang tua calon murid selaku pendaftar (nilai persentase UAT sebesar 85%), dan pihak guru selaku penyeleksi calon murid (nilai persentase UAT sebesar 80%) melalui fitur yang disediakan oleh sistem informasi, yakni fitur mendaftar murid, dan fitur memverifikasi pendaftaran.
2. Berdasarkan hasil pengujian *User Acceptance Testing* yang diberikan kepada pihak guru selaku penilai murid (nilai persentase UAT sebesar 90%), proses penilaian dapat dipermudah dengan menggunakan Sistem Informasi Manajemen Akademik Sekolah Dasar Negeri Wates Kabupaten Kediri, melalui fitur memasukkan detail nilai.

### 7.2 Saran

Saran yang disampaikan untuk penelitian selanjutnya ialah:

1. Untuk penelitian selanjutnya, sistem dapat dikembangkan dengan dengan penambahan fitur, serta proses penilaian untuk kurikulum K13.
2. Diperlukannya panduan penggunaan sistem informasi manajemen akademik sekolah dasar bagi para civitas sekolah ataupun pelatihan sehingga lebih mengetahui bagaimana cara menggunakan sistem yang telah dikembangkan.

## DAFTAR PUSTAKA

- 1keydata, 2017. *Physical Data Model*. [Online] Tersedia di: <http://www.1keydata.com/datawarehousing/physical-data-model.html> [Diakses 6 September 2017].
- Bittner, K. & Spence, I., 2003. *Use Case Modeling*. United States : Pearson Education Inc.
- Booch, et. al., 2005. *The Unified Modeling Language User Guide*. 2<sup>nd</sup> Edition. United States : Pearson Education Inc.
- Cunningham, W., 2001. *Principles Behind the Agile Manifesto*. Tersedia di <<http://agilemanifesto.org/principles.html>> [Diakses 1 September 2017]
- Ependi, U., 2012. *Pengembangan E-Trace Alumni Dengan Menggunakan Pendekatan Metode Agile*. Palembang.
- Homes, B., 2012. *Fundamentals of Software Testing*. London : ISTE.
- Object Management Group (OMG), 2011. *OMG Object Mangement Group*. [Online] Available at: <http://www.omg.org/spec/BPMN/2.0/PDF/> [Diakses 2 Agustus 2017].
- O'Brien, J. A. & Marakas, G. M., 2013. *Introduction to Information Systems*. 16<sup>th</sup> Edition. United States.
- Pressman, R. S., 2010. *Software Engineering A Practitioner's Approach*. 7<sup>th</sup> Edition. New York : McGraw-Hill.
- Shafirna, B. V., 2017. *Analisis Dan Perancangan Sistem Informasi Manajemen Berbasis Website (Studi Kasus KB & TK Permata Iman Malang)*. S1. Universitas Brawijaya.
- Shalahuddin, M. & S, R. A., 2015. *Rekayasa Perangkat Lunak : Terstruktur dan Berorientasi Obyek*. s.l.:Informatika.
- Sommerville, I., 2011. *Software Engineering*. 9<sup>th</sup> Edition. United States of America : Pearson Education Inc.
- Sugiarti, D. & Wardati, I. U., 2012. *Sistem Informasi Akademik Sekolah Dasar Al-Muhajirin Barehan Sidoharjo Pacitan*. Universitas Surakarta. Jawa Tengah.
- Suharjo, 2006. *Mengenal Pendidikan Sekolah Dasar: Teori dan Praktek*. Jakarta: Depdiknas.
- Suhartono, Suparlan, 2008. *Wawasan Pendidikan : Sebuah Pengantar Pendidikan*. Yogyakarta: Ar-Ruzzmedia
- Vliet, H. V., 2007. *Software Engineering: Principles and Practice*. Wiley.
- Weske, M., 2007. *Business Process Management Concept, Languages, Architertures*. New York: Springer-Verlag Berlin Heidelberg.



## LAMPIRAN A DOKUMENTASI WAWANCARA

### A.1 Wawancara dengan Kepala SDN Wates

#### Hasil Wawancara

Narasumber : Kepala SDN Wates

Lokasi Wawancara : SDN Wates, Kecamatan Wates, Kabupaten Kediri.

No	Pertanyaan	Domain Objective	Jawaban
1	Apa saja yang dikerjakan (aktivitas / proses bisnis) akademik di SDN Wates?		Registrasi murid baru. Pendataan alumni. Kurikulum KTSP dan K13. PBM (Proses belajar mengajar) diantaranya membuat Silabus, Program, RPP, KBM, Evaluasi, Penilaian. Ujian ada UTS 1 dan 2, UAS, UKK untuk kelas 1 sampai 5, UAS untuk kelas 6. Evaluasi hasil belajar. Undangan kegiatan sekolah.
2	Bagaimana mengerjakannya (aktivitas / proses bisnis) ?	Menganalisis masalah yang terjadi	Registrasi : orang tua / wali datang ke sekolah untuk mengambil formulir, dan diisi datanya. Setelah data lengkap, formulir diserahkan kembali ke guru untuk diseleksi. Jika memenuhi syarat, maka calon murid (pendaftar) akan direkomendasikan, jika tidak maka tidak direkomendasikan. Hasil dari seleksi akan diumumkan oleh petugas TU melalui papan pengumuman yang dipasang pada tanggal yang telah ditentukan. Pendataan alumni : Petugas TU menyiapkan formulir pendataan alumni, kemudian menghubungi alumni untuk datang ke sekolah guna mengisi formulir pendataan alumni. Formulir pendataan alumni kemudian diarsip. Kurikulum: dikerjakan mulai April sampai Juni mulai review, evaluasi diri, menetapkan rancangan kurikulum, penetapan kurikulum. Silabus dan program dikerjakan

		awal tahun. RPP dikerjakan sesuai pembelajaran (tematik). Evaluasi dilakukan setiap 1 pokok bahasan untuk KTSP, dan setiap KD untuk K13. K13 menggunakan aplikasi, sedangkan KTSP manual tulis biasa. Untuk rapotan membutuhkan anggaran dana untuk membuat undangan ke wali murid (563 murid). Pembuatan soal UTS, UAS, UKK ada timnya sendiri untuk 1 kecamatan. Evaluasi murid : Guru memberikan ujian untuk dikerjakan murid, dan kemudian dinilai. Setelah dinilai, hasil pekerjaan murid dikembalikan. Hasil penilaian direkap dan dilaporkan ke kepala sekolah. Keseluruhan nilai yang telah diolah akan dibagikan pada waktu penerimaan rapot. Undangan kegiatan sekolah : guru menyiapkan undangan kegiatan yang akan diberikan orang tua / wali melalui murid. Ketika undangan sudah tersampaikan, orang tua / wali datang menghadiri kegiatan.
3	Siapa sajakah yang terlibat dalam mengerjakannya?	Guru, kepala sekolah, murid, orang tua / wali murid, komite, petugas TU tenaga kependidikan lainnya
4	Dimana mengerjakan aktivitas tersebut?	Untuk kegiatan proses belajar mengajar, dilakukan di sekolah
5	Perluakah dikerjakan?	Sangat perlu, dan harus didukung dengan sarana dan prasarana yang memadai
6	Apakah prosedur kegiatan akademik telah dikerjakan dengan baik? Apakah terdapat kendala?	Terdapat beberapa kendala seperti sarana & prasarana yang kurang lengkap, kurangnya laboratorium (IPA, Bahasa), terbatasnya sarana penyebaran informasi. Tidak tersampainya undangan kegiatan sekolah, proses registrasi termasuk seleksi yang menyulitkan pendataan dan pengelolaan

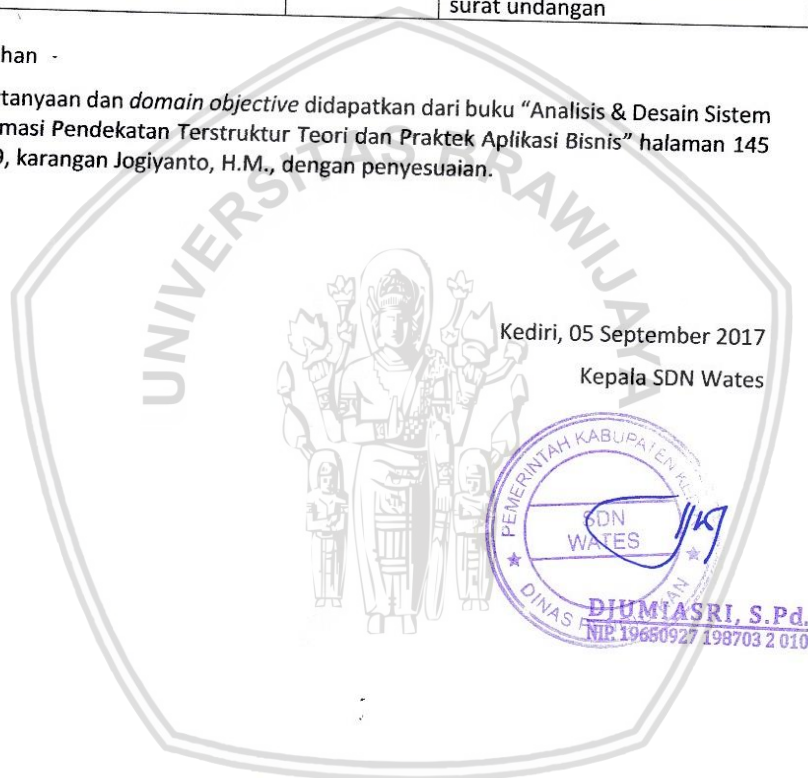
			berkas, karena masih dikerjakan secara manual. Kendala lainnya juga terdapat pada sulitnya pendataan alumni.
7	Bagaimana penyampaian informasi dalam kegiatan akademik selama ini? *	Menganalisis distribusi pekerjaan	Penyampaian informasi menggunakan surat undangan untuk wali murid, membuat poster untuk kegiatan, dan menggunakan facebook SDN Wates. Untuk informasi khusus biasanya disampaikan guru kelas kepada murid.
8	Apakah tugas dan tanggung jawab sudah didefinisikan dan diterapkan dengan jelas pada akademik?		Sudah, dilengkapi surat tugas untuk guru dan tenaga kependidikan
9	Apakah masing-masing kegiatan telah mencapai sasarannya?		Sudah, tetapi pencapaian belum maksimal, masih terdapat kendala wali murid, alumni, sarana & prasarana yang belum lengkap
10	Apakah terjadi aktivitas yang tumpang tindih?	Menganalisis pengukuran pekerjaan	Tidak ada
11	Seberapa perlu hasil dari tiap-tiap aktivitas?		Sangat perlu
12	Apakah terdapat standarisasi kinerja yang baik dan selalu dimutakhirkan?		Ada, mengacu pada DAPODIK
13	Apakah operasi-operasi telah direncanakan dengan baik dan terkendali?	Menganalisis keandalan	Perencanaan baik, namun ada beberapa hambatan
14	Seberapa perlu dokumen-dokumen yang ada?		Sangat perlu, seperti dokumen kegiatan, pembelajaran & raport, dll
15	Apakah masing-masing dokumen dirancang untuk penggunaan yang efektif?	Menganalisis dokumen	Iya
16	Apakah tembusan-tembusan dari dokumen perlu? Jika iya, kemana saja tembusan ditujukan?		Iya, dokumen keuangan dan kegiatan ditembuskan pada Dinas, dokumen DAPODIK ditembuskan ke Dinas Pendidikan Daerah Nasional
17	Dapatkah laporan-laporan dipersiapkan dengan mudah dari file dan dokumen-dokumen yang ada?	Menganalisis laporan	Harapannya begitu



18	Apakah terdapat duplikasi di file, catatan-catatan dan laporan-laporan?		Semua di duplikatkan, seperti raport, ijazah, laporan keuangan, laporan pendidikan, laporan kegiatan
19	Apakah file mudah diakses dan dimutakhirkan?		Mudah, seperti DAPODIK bisa diupdate online
20	Apakah peralatan-peralatan pengolahan data telah digunakan dengan efektif?	Menganalisis teknologi	Belum
21	Apakah sudah ada sistem untuk membantu proses pengelolaan (aktivitas / proses bisnis) ? *	Menganalisis tingkat kebutuhan sistem *	Untuk pengelolaan laporan keuangan sudah ada sistem onlinenya, namun untuk penyebaran informasi kepada wali murid menggunakan facebook dan surat undangan

\* = tambahan -

Daftar pertanyaan dan *domain objective* didapatkan dari buku "Analisis & Desain Sistem Informasi Pendekatan Terstruktur Teori dan Praktek Aplikasi Bisnis" halaman 145 – 149, karangan Jogiyanto, H.M., dengan penyesuaian.



## LAMPIRAN B DOKUMENTASI UAT

### User Acceptance Testing – Mengisi Formulir Pendaftaran Murid (Studi Pada SDN Wates Kabupaten Kediri)

Nama : Abdul Rachman  
Jabatan : Orang tua  
Tanggal Pengujian : 9 April 2018

Kuesioner ini digunakan untuk mengetahui apakah masalah sebelum adanya sistem informasi dapat teratasi setelah adanya sistem informasi.

Lingkari pada jawaban penilaian yang anda pilih.

No	Nama Pengujian	Penilaian				
		Sangat Baik	Baik	Netral	Kurang	Sangat Kurang
1	Apakah dengan adanya sistem informasi manajemen akademik dapat membantu mempermudah orang tua untuk mendaftarkan anaknya (calon murid)?	Sangat Membantu	Membantu	Sama Saja	Kurang Membantu	Tidak Membantu

Responden

  
(Abdul Rachman)

**User Acceptance Testing – Mengisi Formulir Pendaftaran Murid  
(Studi Pada SDN Wates Kabupaten Kediri)**

Nama : Diah  
 Jabatan : Orang Tua  
 Tanggal Pengujian : 9 April 2018

Kuesioner ini digunakan untuk mengetahui apakah masalah sebelum adanya sistem informasi dapat teratasi setelah adanya sistem informasi.

Lingkari pada jawaban penilaian yang anda pilih.

No	Nama Pengujian	Penilaian				
		Sangat Baik	Baik	Netral	Kurang	Sangat Kurang
1	Apakah dengan adanya sistem informasi manajemen akademik dapat membantu mempermudah orang tua untuk mendaftarkan anaknya (calon murid)?	Sangat Membantu	Membantu	Sama Saja	Kurang Membantu	Tidak Membantu

Responden

  
 ( Diah )

## User Acceptance Testing – Mengisi Formulir Pendaftaran Murid (Studi Pada SDN Wates Kabupaten Kediri)

Nama : Mulyani  
Jabatan : Orang tua  
Tanggal Pengujian : 9 April 2018

Kuesioner ini digunakan untuk mengetahui apakah masalah sebelum adanya sistem informasi dapat teratasi setelah adanya sistem informasi.

Lingkari pada jawaban penilaian yang anda pilih.

No	Nama Pengujian	Penilaian				
		Sangat Baik	Baik	Netral	Kurang	Sangat Kurang
1	Apakah dengan adanya sistem informasi manajemen akademik dapat membantu mempermudah orang tua untuk mendaftarkan anaknya (calon murid)?	Sangat Membantu	Membantu	Sama Saja	Kurang Membantu	Tidak Membantu

Responden

  
 ( Mulyani )



**User Acceptance Testing – Mengisi Formulir Pendaftaran Murid  
(Studi Pada SDN Wates Kabupaten Kediri)**

Nama : M. Abbas  
 Jabatan : Orang tua  
 Tanggal Pengujian : 9 April 2018

Kuesioner ini digunakan untuk mengetahui apakah masalah sebelum adanya sistem informasi dapat teratasi setelah adanya sistem informasi.

Lingkari pada jawaban penilaian yang anda pilih.

No	Nama Pengujian	Penilaian				
		Sangat Baik	Baik	Netral	Kurang	Sangat Kurang
1	Apakah dengan adanya sistem informasi manajemen akademik dapat membantu mempermudah orang tua untuk mendaftarkan anaknya (calon murid)?	Sangat Membantu	Membantu	Sama Saja	Kurang Membantu	Tidak Membantu

Responden



( M. Abbas )

### User Acceptance Testing – Melakukan Verifikasi Pendaftaran (Studi Pada SDN Wates Kabupaten Kediri)

Nama : HARTANTO  
 Jabatan : Guru kelas, I  
 Tanggal Pengujian : 9 April 2018

Kuesioner ini digunakan untuk mengetahui apakah masalah sebelum adanya sistem informasi dapat teratasi setelah adanya sistem informasi.

Lingkari pada jawaban penilaian yang anda pilih.

No	Nama Pengujian	Penilaian				
		Sangat Baik	Baik	Netral	Kurang	Sangat Kurang
1	Apakah dengan adanya sistem informasi manajemen akademik dapat membantu mempermudah guru dalam menyeleksi pendaftaran calon murid?	Sangat Membantu	Membantu	Sama Saja	Kurang Membantu	Tidak Membantu

Responden

  
 (HARTANTO)

### User Acceptance Testing – Melakukan Verifikasi Pendaftaran (Studi Pada SDN Wates Kabupaten Kediri)

Nama : Rin Harini  
 Jabatan : Guru Kelas 1  
 Tanggal Pengujian : 9 April 2018

Kuesioner ini digunakan untuk mengetahui apakah masalah sebelum adanya sistem informasi dapat teratasi setelah adanya sistem informasi.

Lingkari pada jawaban penilaian yang anda pilih.

No	Nama Pengujian	Penilaian				
		Sangat Baik	Baik	Netral	Kurang	Sangat Kurang
1	Apakah dengan adanya sistem informasi manajemen akademik dapat membantu mempermudah guru dalam menyeleksi pendaftaran calon murid?	Sangat Membantu	Membantu	Sama Saja	Kurang Membantu	Tidak Membantu

Responden

  
 ( Rin Harini )

### User Acceptance Testing – Memasukkan Detail Nilai (Studi Pada SDN Wates Kabupaten Kediri)

Nama : HARTANTO  
 Jabatan : Guru Kelas I  
 Tanggal Pengujian : 9 April 2018

Kuesioner ini digunakan untuk mengetahui apakah masalah sebelum adanya sistem informasi dapat teratasi setelah adanya sistem informasi.

Lingkari pada jawaban penilaian yang anda pilih.

No	Nama Pengujian	Penilaian				
		Sangat Baik	Baik	Netral	Kurang	Sangat Kurang
1	Apakah dengan adanya sistem informasi manajemen akademik dapat membantu mempermudah guru dalam memberikan nilai murid?	Sangat Membantu	Membantu	Sama Saja	Kurang Membantu	Tidak Membantu

Responden

(Hartanto)

**User Acceptance Testing – Memasukkan Detail Nilai  
(Studi Pada SDN Wates Kabupaten Kediri)**

Nama : Dinda Retno Nugraha  
 Jabatan : Guru Kelas 2  
 Tanggal Pengujian : 9 April 2018

Kuesioner ini digunakan untuk mengetahui apakah masalah sebelum adanya sistem informasi dapat teratasi setelah adanya sistem informasi.

Lingkari pada jawaban penilaian yang anda pilih.

No	Nama Pengujian	Penilaian				
		Sangat Baik	Baik	Netral	Kurang	Sangat Kurang
1	Apakah dengan adanya sistem informasi manajemen akademik dapat membantu mempermudah guru dalam memberikan nilai murid?	Sangat Membantu	Membantu	Sama Saja	Kurang Membantu	Tidak Membantu

Responden



(Dinda Retno Nugraha)



### User Acceptance Testing – Memasukkan Detail Nilai (Studi Pada SDN Wates Kabupaten Kediri)

Nama : Zaenal Arifin  
 Jabatan : Guru kelas 3  
 Tanggal Pengujian : 9 April 2018

Kuesioner ini digunakan untuk mengetahui apakah masalah sebelum adanya sistem informasi dapat teratasi setelah adanya sistem informasi.

Lingkari pada jawaban penilaian yang anda pilih.

No	Nama Pengujian	Penilaian				
		Sangat Baik	Baik	Netral	Kurang	Sangat Kurang
1	Apakah dengan adanya sistem informasi manajemen akademik dapat membantu mempermudah guru dalam memberikan nilai murid?	Sangat Membantu	Membantu	Sama Saja	Kurang Membantu	Tidak Membantu

Responden

  
 (Zaenal Arifin)

### User Acceptance Testing – Memasukkan Detail Nilai (Studi Pada SDN Wates Kabupaten Kediri)


Nama : Asri Yuniarti  
 Jabatan : Guru Kelas 4  
 Tanggal Pengujian : 9 April 2018

Kuesioner ini digunakan untuk mengetahui apakah masalah sebelum adanya sistem informasi dapat teratasi setelah adanya sistem informasi.

Lingkari pada jawaban penilaian yang anda pilih.

No	Nama Pengujian	Penilaian				
		Sangat Baik	Baik	Netral	Kurang	Sangat Kurang
1	Apakah dengan adanya sistem informasi manajemen akademik dapat membantu mempermudah guru dalam memberikan nilai murid?	Sangat Membantu	Membantu	Sama Saja	Kurang Membantu	Tidak Membantu

Responden

  
 (ASRI YUNIARTI)



## LAMPIRAN C DOKUMENTASI VALIDASI BPMN

### VALIDASI BPMN (TO-BE)

Nama Validator : Ibu Djumiasri

Jabatan : Kepala SDN Wates

Beri tanda centang (✓) pada kriteria yang dipilih.

No	BPMN To-Be	Validitas	
		Valid	Tidak Valid
1	Proses Bisnis Registrasi Usulan	✓	
2	Proses Bisnis Mengevaluasi Murid Usulan	✓	

Kediri, 18 Mei 2018

Validator



**DJUMIASRI, S.Pd.**  
NIP. 19650927 198703 2 010